Maths

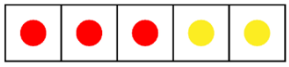
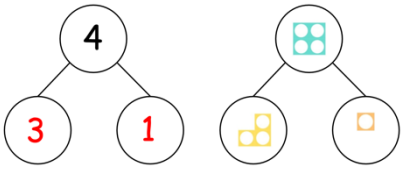
Calculation Policy

Addition

2024

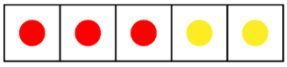
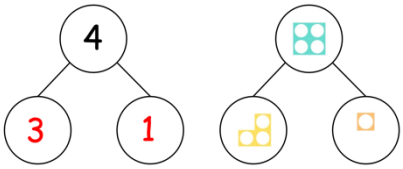
EYFS

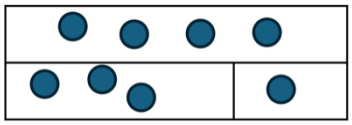
- I know numbers can be partitioned in different ways.

1. Numicon 
2. Tens/fives frame 
3. Part part whole model 

- I know a number represents how many objects are in a set
- I can subitise up to 5 confidently
- I know the number after a given number is one more

- I know all the different compositions of 1
- I know all the different compositions of 2
- I know all the different compositions of 3
- I know all the different compositions of 4
- I know all the different compositions of 5

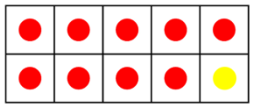
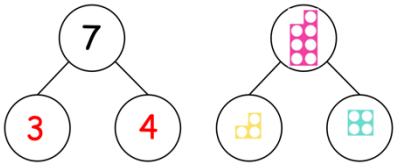
1. Numicon 
2. Tens/five frames 
3. Part part whole model (using counters, Numicon and numbers) 

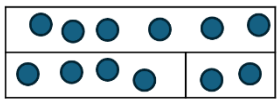
4. Bar model (using counters, Numicon and numbers) 

- I can recall number bonds to 5

- I know all the different compositions of 6
- I know all the different compositions of 7
- I know all the different compositions of 8

- I know all the different compositions of 9
- I know all the different compositions of 10

1. Numicon 
2. Tens frame 
3. Part part whole model (using counters, Numicon and numbers) 

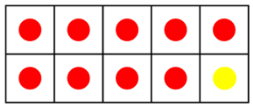
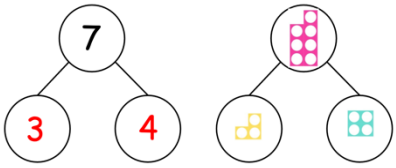
4. Bar model (using counters, Numicon and numbers) 

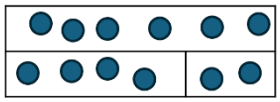
- I can recall some number bonds to 10
  - 5 + 5
  - 9 + 1
  - 10 + 0
  - 2 + 8

- I can double numbers from 0-5

Year One

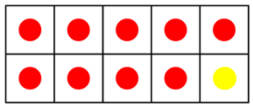
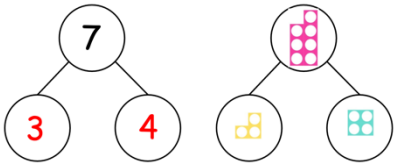
- I know that numbers can be partitioned in to two groups

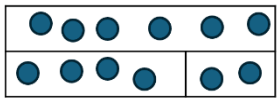
1. Numicon 
2. Tens frame 
3. Part part whole model (using counters, Numicon and numbers) 

4. Bar model (using counters, Numicon and 

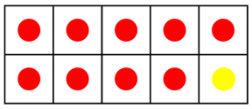
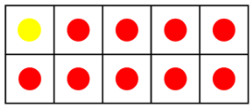
numbers)

- I can recall my number bonds to 10
- I know what combining two or more parts makes a whole set (*up to 10*)

1. Numicon 
2. Tens frame 
3. Part part whole model (using counters, Numicon and numbers) 

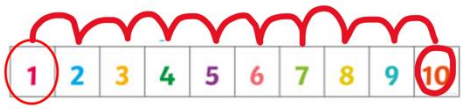
4. Bar model (using counters, Numicon and numbers) 

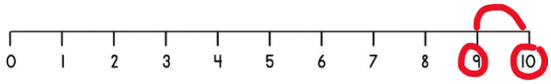
- I know that addition is commutative

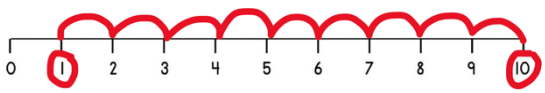
1. Tens frame  

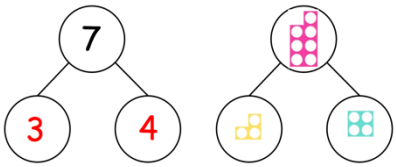
9 add 1 equals 10 1 add 9 equals 10

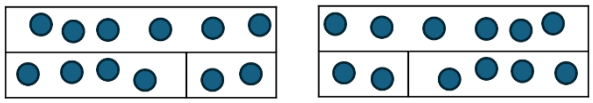
2. Number track 



3. Number line 

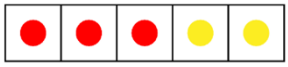
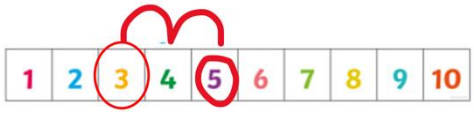


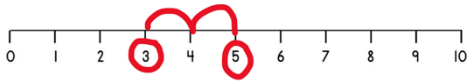
4. Part part whole model 

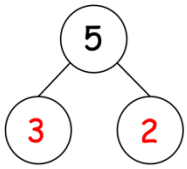
5. Bar model 

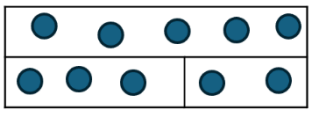
- I can add to an existing group to find the total using 'first', 'then' and 'now'

  E.g.) There are 3 children on the swings, then 2 more join. Now there are 5 children.

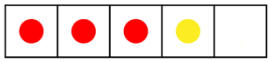
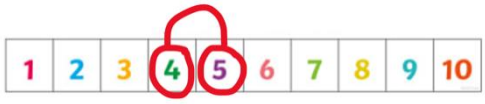
1. Numicon 
2. Tens frame 
3. Number track 

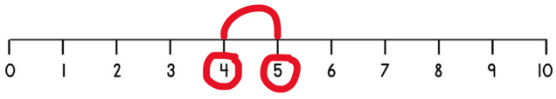
4. Number line 

5. Part part whole model 

6. Bar model 

- I know the equals sign represents the total amount
- I know the addition sign means add
- I know that a number is one more than a given number
- I can find one more and know it is the next number in the count

1. Tens frame 
2. Number track 

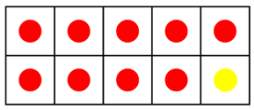
3. Number line 

4. Using mathematic symbols eg → 4 + 1 = 5

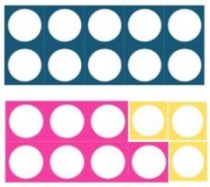
- I know when 0 is added to a number it stays the same
  E.g.) 5 + 0 = 5 (0 is added to 5)
  0 + 5 = 5 (5 is added to 0)

- I know that 10 can be partitioned into pairs

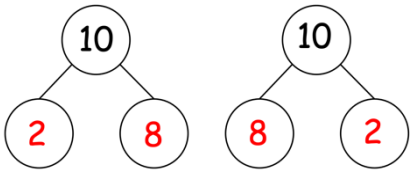
1. Tens frame
   9 and 1
   9 + 1 = 10
   1 and 9
   1 + 9 = 10



2. Numicon
   7 and 3
   7 + 3 = 10
   3 and 7
   3 + 7 = 10



3. Part part whole model
   2 and 8
   2 + 8 = 10
   8 and 2
   8 + 2 = 10

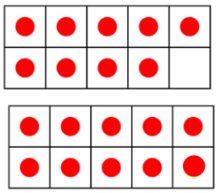


- I can recall number bonds to 10
  Without resources

- I can find 10 more than a number

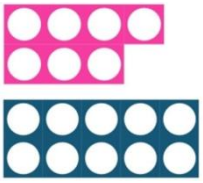
1. Tens frame

   9 + 10 = 19



2. Numicon

   7 + 10 = 17

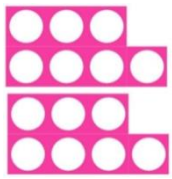


3. Bead string

4. Number square

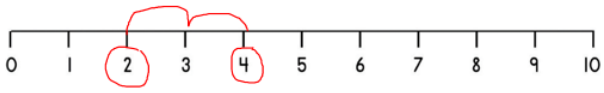
- I can double numbers up to 10

1. Numicon



2. Number tracks

3. Number lines



- I can use known facts to add 1 digit and 1 digit (e.g. doubling)

E.g. 6 + 7 =

6 + 6 = 12 then + 1 = 13

1. Numicon
What's the difference?

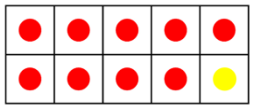
2. Bar model

13	
6	7
6	6 | 1

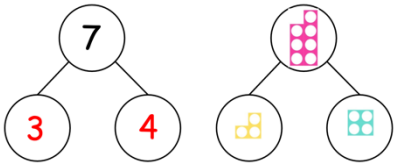
- I can partition a single digit in different ways

1. Numicon

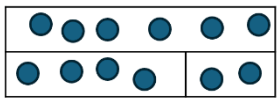
2. Tens frame



3. Part part whole model
(using counters, Numicon and numbers)



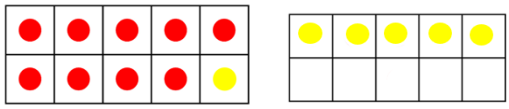
4. Bar model
(using counters, Numicon and numbers)



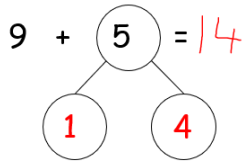
- I can begin to bridge through 10

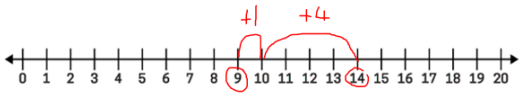
E.g. 9 + 5 =

1. Tens frame



2. Number line using part part whole model



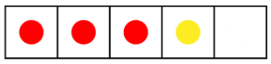


- Partition the smaller number to make a number bond to 10
- Add the 1 to get to 10
- Add the remaining 4

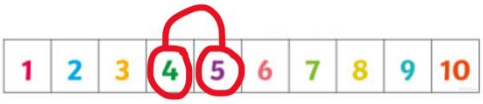
Year Two

- I know the equals sign represents the total amount
- I know the addition sign means add
- I can find one more and know it is the next number in the count

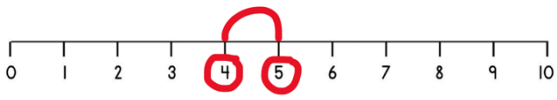
1. Tens frame



2. Number track



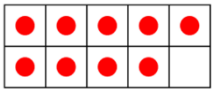
3. Number line



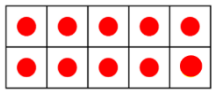
4. Using mathematic symbols eg → 4 + 1 = 5

- I can find 10 more than a number

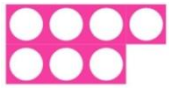
1. Tens frame



9 + 10 = 19



2. Numicon



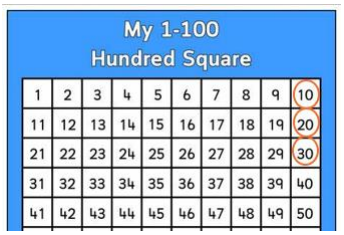
7 + 10 = 17

3. Bead string

4. Number square

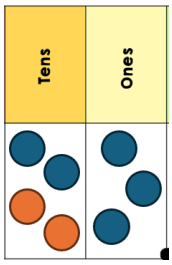
- I can add a multiple of 10 to a 1 digit and 2 digit number

1. Number square



2. Place value chart

23 + 20 = 43

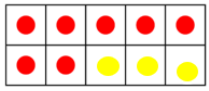


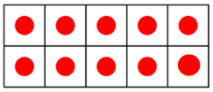
3. 32 + 40 = 72
Most of the time, only the tens digit changes and the ones stays the same.

Children also need to go beyond 100 – e.g. 82 + 20 = 102
Both the hundreds tens column has changed.

- I can use addition facts to 10 to derive addition facts to 20
E.g. 17 + 3 = → 7 + 3 = 10 so 17 + 3 = 20

1. Tens frame





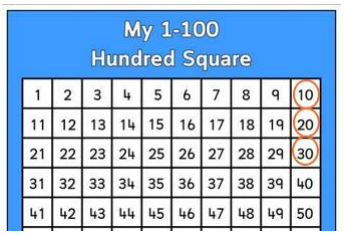
2. Multi-link cubes

3. Bar model

20		
17		3
10	7	3

- I can count in multiples of 10

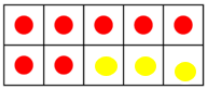
1. Number square



1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50

- I can use addition facts to find addition facts to 100 (using multiples of 10)

1. Tens frame
   Use ones counters and then replace them with tens counters



2. Transfer to written methods

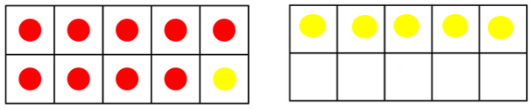
   50 + 30 = 80

   5 + 3 = 8

- I can begin to bridge through 10

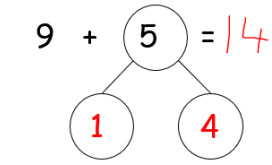
E.g. 9 + 5 =

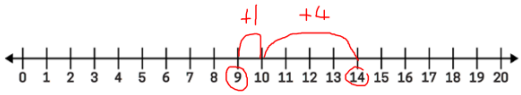
3. Tens frame



4. Number line using part part whole model

   - Partition the smaller number to make a number bond to 10
   - Add the 1 to get to 10
   - Add the remaining 4

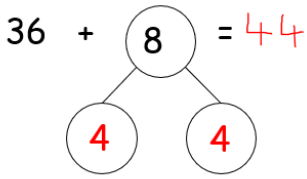


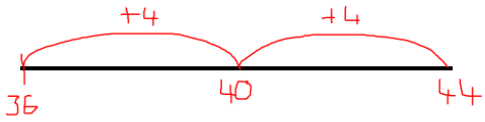


- I can bridge through 10 when adding 2 digit and 1 digit numbers

  E.g. 36 + 8 =

1. Number line using part part whole model



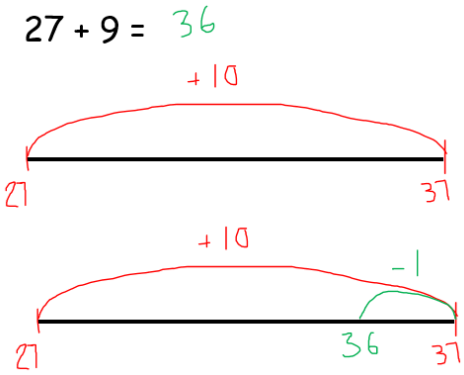


- I can add 9 using compensating

  E.g. 27 + 9 =

1. Tens frame
2. Bead string
3. Number line

   - *First add 10*
   - *Then subtract 1*

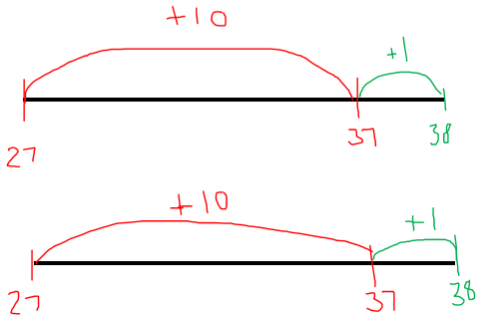


- I can add 11 using compensating

  E.g. 27 + 11 =

  1. Tens frame
  2. Bead string
  3. Number line
     - *First add 10*
     - *Then add 1*

27 + 11 = 38

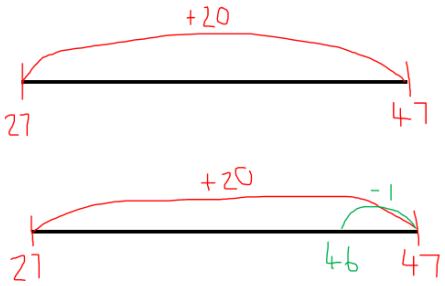


- I can add 19 using compensating

  E.g. 27 + 19 =

  1. Tens frame
  2. Bead string
  3. Number line
     - *First add 20*
     - *Then subtract 9*

27 + 19 = 46



- I can add 21 using compensating

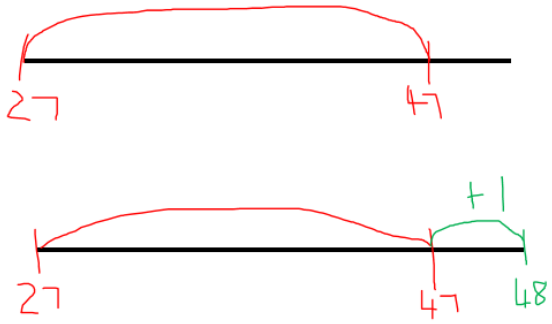
  E.g. 27 + 19 =

  1. Tens frame
  2. Bead string

3. Number line

27 + 21 = 48

- *First add 20*
- *Then add 1*



- I can use known facts to add digit and 1 digit (e.g. doubling)

  E.g. 6 + 7 =

  6 + 6 = 12 then + 1 = 13

  1. Numicon
     What's the difference?

  2. Bar model

13	
6	7
6	6 | 1

- I can use known facts to add 2 digit numbers using doubling

  E.g. 14 + 15 =

  14 + 14 = 28 then + 1 = 29

  1. Bar model

29		
14	15	
14	14	1

- I can recall number bonds to 10

- I can add three 1 digit numbers using reordering when two are number bonds

E.g. 3 + 5 + 7 =

3 + 5 + 7 =

3 + 7 + 5 =

10 + 5 = 15

- I can add three 1 digit numbers using reordering and appropriate strategies ***(When there are no number bonds)***

E.g. 6 + 7 + 1 =

6 + 7 + 1=

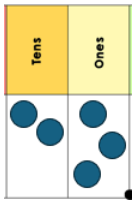
6 + 1 + 7 =

7 + 7 = 14 (doubling)

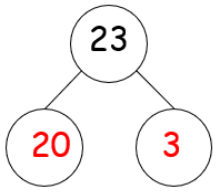
GD → can spot patterns within the numbers and use doubling etc
EXP and below → just add the numbers

- I can partition 2 digit numbers into tens and ones

1. Place value grid with counters and digits



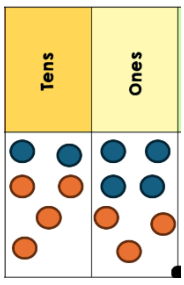
2. Part part whole model



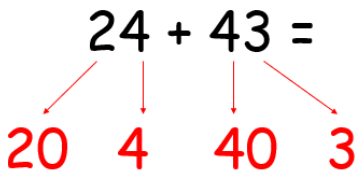
- I can add two 2 digit numbers using partitioning without crossing a boundary

1. Place value grid with counters and digits

24 + 43 =



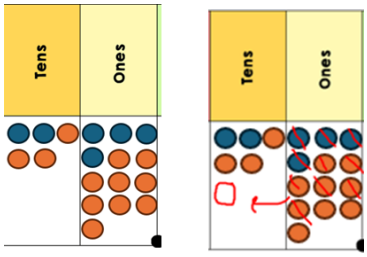
2. Partitioning



20 + 40 = 60
4 + 3 = 7
60 + 7 = 67

- I can add two 2 digit numbers using partitioning when crossing a boundary

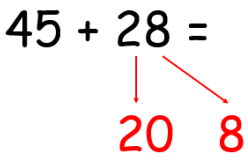
1. Place value chart and counters



    1. Put in the largest number
    2. Add the tens
    3. Add the ones
    4. Make a 10 using the ones and put into the tens column

2. Partitioning

    1. Start with the largest number
    2. Add the tens
    3. Add the ones (may need to use another method to support eg. Bridging, doubling etc)

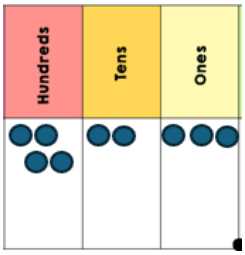


45 + 20 = 65
65 + 8 = 73 (use bridging)

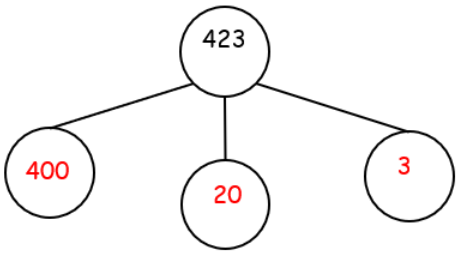
Year Three

- I can partition 3 digit numbers

1. Place value grid with counters and digits

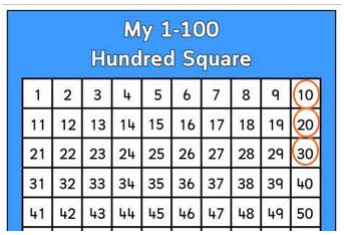


2. Part part whole model



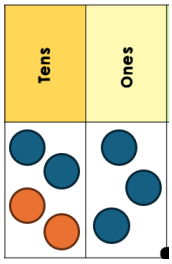
- I can add a multiple of 10 to a 1 digit and 2 digit number

1. Number square



2. Place value chart

23 + 20 = 43



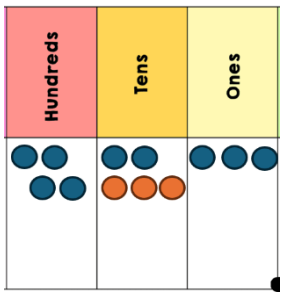
3. 32 + 40 = 72
Most of the time, only the tens digit changes and the ones stays the same.

Children also need to go beyond 100 – e.g. 82 + 20 = 102
Both the hundreds tens column has changed.

- I can multiples of 10 to 3 digit numbers without crossing a boundary

E.g. 423 + 30 =

1. Place value grid



2.

3.

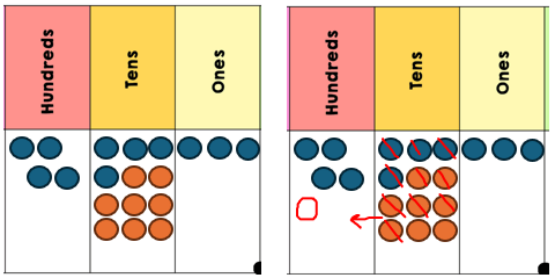
Rule:

1. Underline the tens column (you are adding a multiple of 10)
2. Add them together
3. Everything else stays the same

$$\underline{4}23 + \underline{3}0 = 453$$

- I can add multiples of 10 to 3 digit numbers when crossing a boundary

1. Place value grid



4.

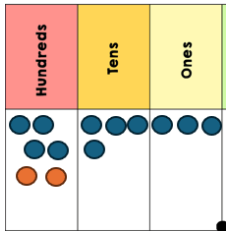
Rule:

1. Underline the tens column (you are adding a multiple of 10)
2. Add them together
3. Exchange by giving the extra '10' to the hundreds

$$\underline{4}43 + \underline{8}0 = 523$$

- I can add multiples of 100 without crossing a boundary

1. Place value grid



1.

Rule:

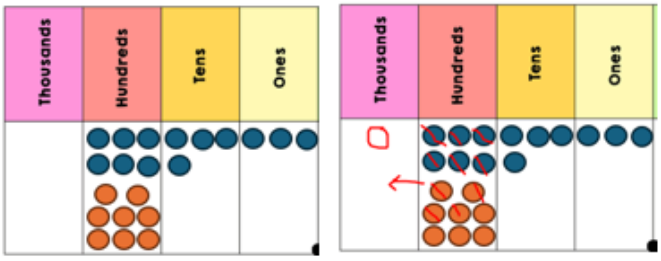
1. Underline the hundreds column (you are adding a multiple of 100)
2. Add them together
3. Everything else stays the same

$$\underline{4}43 + \underline{2}00 = 643$$

$$\underline{6}43 + \underline{8}00 = 1443$$

- I can add multiples of 100 to 3 digit numbers when crossing a boundary

    1. Place value grid

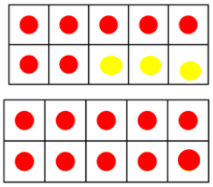


    1.

       Rule:
       1. Underline the tens column (you are adding a multiple of 100)
       2. Add them together
       3. Exchange by giving the extra '100' to the thousands column

- I can use addition facts to 10 to derive addition facts to 20
  E.g. 17 + 3 = → 7 + 3 = 10 so 17 + 3 = 20

    1. Tens frame



    2. Multi-link cubes

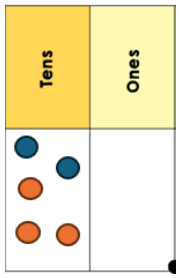
    3. Bar model

20		
17		3
10	7	3

- I can add two 2 digit numbers using known facts

  E.g. 20 + 30 = → 2 + 3 = 5 so 20 + 30 = 50

1. Place value grid and counters



2. Explain why we put the 0s back on (each number is ten times bigger so the answer needs to be ten times bigger)

- I can add two 3 digit numbers using known facts

  E.g. 200 + 300 = → 2 + 3 = 5 so 200 + 300 = 500

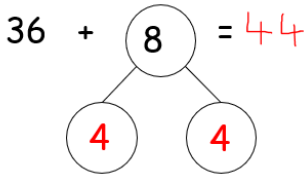
1. Place value grid and counters

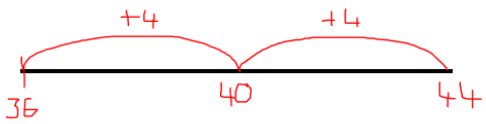
2. Explain why we put the 0s back on (each number is one hundred times bigger so the answer needs to be ten one hundred bigger)

- I can bridge through 10 when adding 2 digit and 1 digit numbers

  E.g. 36 + 8 =

1. Number line using part part whole model

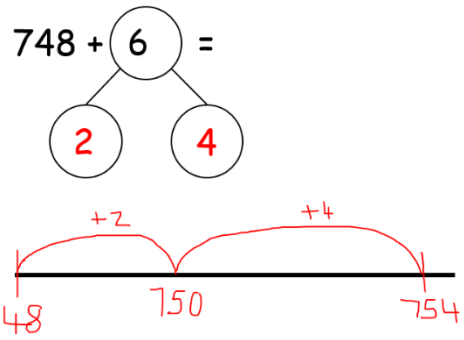




- I can add a 3 digit and 1 digit number using bridging

  E.g. 748 + 6 =

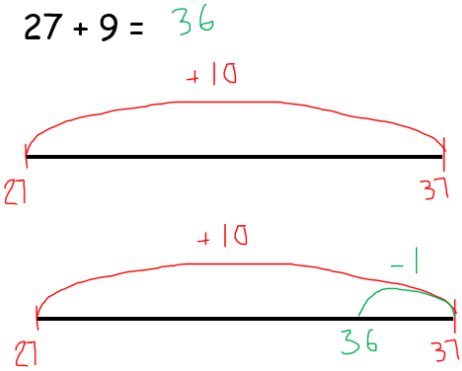
1. Number line using part part whole model



- I can add 9 using compensating

  E.g. 27 + 9 =

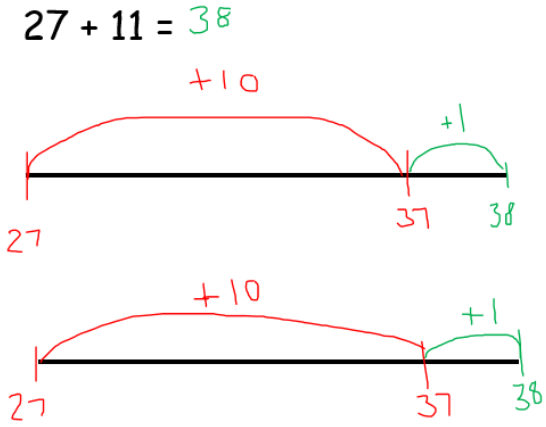
  1. Tens frame
  2. Bead string
  3. Number line
     - *First add 10*
     - *Then subtract 1*



- I can add 11 using compensating

  E.g. 27 + 11 =

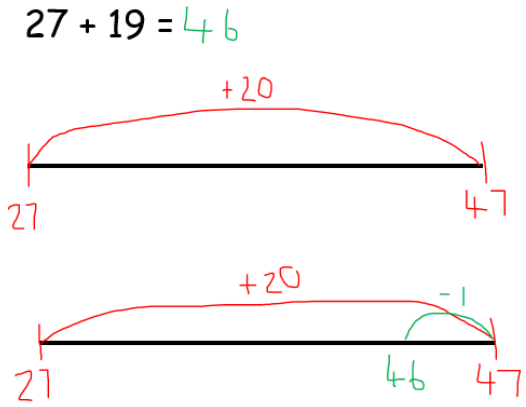
  1. Tens frame
  2. Bead string
  3. Number line
     - *First add 10*
     - *Then add 1*



- I can add 19 using compensating

E.g. 27 + 19 =

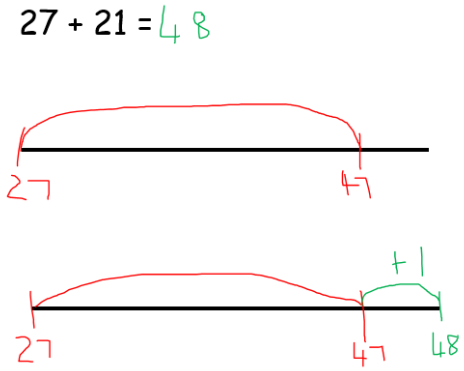
1. Tens frame
2. Bead string
3. Number line
   - *First add 20*
   - *Then subtract 9*



- I can add 21 using compensating

E.g. 27 + 19 =

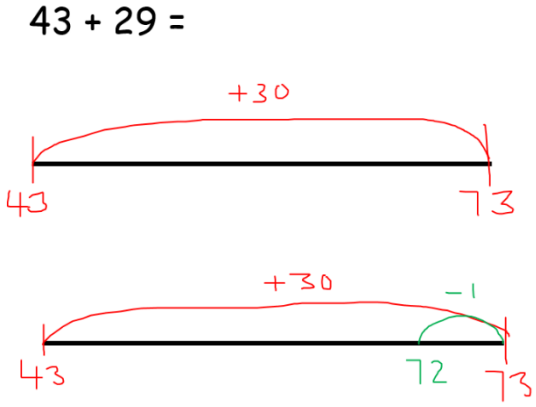
1. Tens frame
2. Bead string
3. Number line
   - *First add 20*
   - *Then add 1*



- I can add two 2 digit numbers using compensating

E.g. + _9 and + _1
GD → + _7 + _8 + _2 + _3 etc

1. Tens frame
2. Bead string
3. Number line
   - *First add 30*
   - *Then subtract 1*

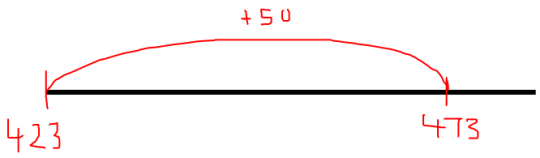


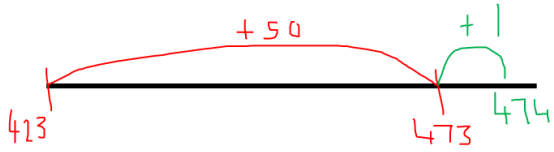
- I can add a 3 digit and 2 digit number using compensating

E.g. + _ _ 9 and + _ _1
GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

1. Tens frame
2. Bead string
3. Number line
   - *First add 50*
   - *Then add 1*

423 + 51 = 474





- I can add two 3 digit numbers using compensating
  Use previous method

- I can use known facts to add 2 digit numbers using doubling

E.g. 14 + 15 =
14 + 14 = 28 then + 1 = 29

1. Bar model

29		
14	15	
14	14	1

- I can add two multiples of 10 using doubling

E.g. 60 + 70 =

1. Bar model

130		
60	70	
60	60	10

- I can add three 1 digit numbers using reordering when two are number bonds

E.g. 3 + 5 + 7 =

3 + 5 + 7 =

3 + 7 + 5 =

10 + 5 = 15

- I can add three 1 digit numbers using reordering and appropriate strategies
  (When there are no number bonds)

  E.g. 6 + 7 + 1 =

  6 + 7 + 1=

  6 + 1 + 7 =

  7 + 7 = 14 (doubling)

- I can add three multiples of 10 using reordering using number bonds

  E.g. 30 + 50 + 70 =

  30 + 50 + 70 =

  30 + 70 + 50 =

  100 + 50 = 150

- I can add three 2 digit and 1 digit numbers using reordering
  Looking for known facts

  E.g. 23 + 14 + 17 =

  23 + 14 + 17 =

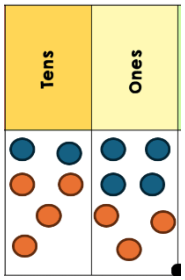
  23 + 17+ 14 =

  40 + 14 = 54 (number bonds)

- I can add two 2 digit numbers using partitioning without crossing a boundary

  1. Place value grid with counters and digits

     24 + 43 =



2. Partitioning

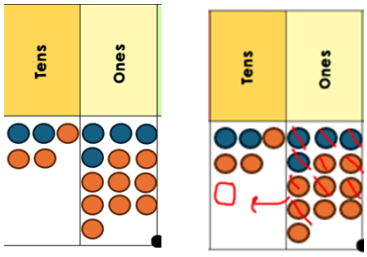
24 + 43 =

20 4 40 3

20 + 40 = 60
4 + 3 = 7
60 + 7 = 67

- I can add two 2 digit numbers using partitioning when crossing a boundary

1. Place value chart and counters



    1. Put in the largest number
    2. Add the tens
    3. Add the ones
    4. Make a 10 using the ones and put into the tens column

2. Partitioning

45 + 28 =

20 8

45 + 20 = 65
65 + 8 = 73 (use bridging)

    1. Start with the largest number
    2. Add the tens
    3. Add the ones (may need to use another method to support eg. Bridging, doubling etc)

- I can add a 3 digit and 1 digit number using partitioning without crossing a boundary

1. Place value grid with counters and digits

354 + 3 =

2. Partitioning

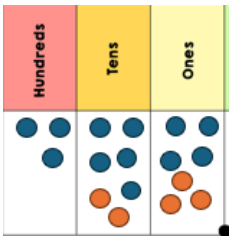
354 + 3 =

300 50 4 3

300
50
4 + 3 = 7
300 + 50 + 7 = 357

- I can add a 3 digit and 2 digit number using partitioning without crossing a boundary

1. Place value grid with counters and digits



354 + 23 =

2. Partitioning

354 + 23 =

300 50 4 20 3

300
50 + 20 = 70
4 + 3 = 7
300 + 70 + 7 = 377

- I can add a 3 digit and 3 digit number using partitioning without crossing a boundary

1. Place value grid with counters and digits

354 + 123 =

2. Partitioning

354 + 123 =

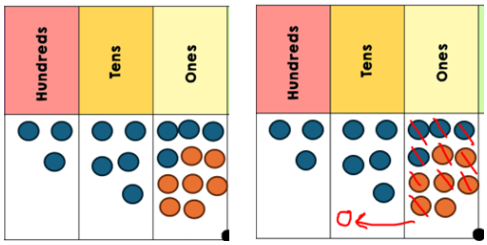
300 50 4 100 20 3

300 + 100 = 400
50 + 20 = 70
4 + 3 = 7
400 + 70 + 7 = 477

- I can add a 3 digit and 1 digit number using partitioning when crossing a boundary

1. Place value chart and counters

    1. Put in the largest number
    2. Add the ones
    3. Make a 10 using the ones and put into the tens column



2. Partitioning

    1. Start with the largest number
    2. Add the ones (may need to use another method to support eg. Bridging, doubling etc)

354 + 7 =

7

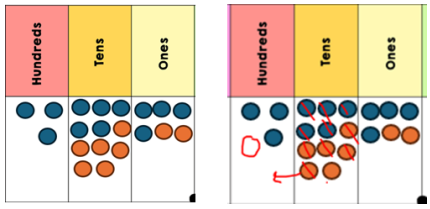
354 + 7 = 361

(use an appropriate method e.g. bridging)

- I can add a 3 digit and 1 digit number using partitioning when crossing a boundary

1. Place value chart and counters

    1. Put in the largest number
    2. Add the ones
    3. Add the tens
    4. Make a 100 using the tens and put into the hundreds column



2. Partitioning

    1. Start with the largest number
    2. Add the ones, tens (may need use another method to support Bridging, doubling etc)

354 + 62 =

60 2

354 + 2 = 356

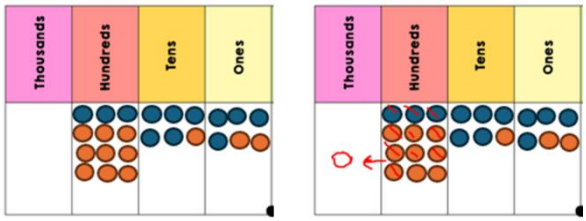
356 + 60 = 416

(use an appropriate method to add)

to eg.

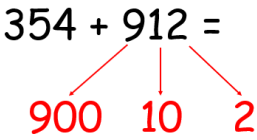
- I can add a 3 digit and 3 digit number using partitioning when crossing a boundary

1. Place value chart and counters



    1. Put in the largest number
    2. Add the ones
    3. Add the tens
    4. Add the hundreds
    5. Make a 100 using the tens and put into the hundreds column

2. Partitioning



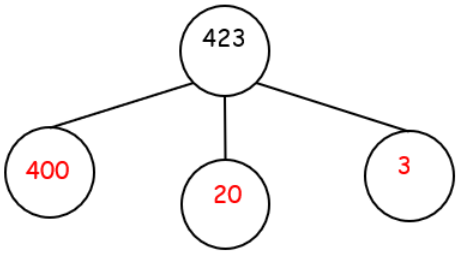
    1. Start with the largest number
    2. Add the ones, tens, hundreds (may need to use another method to support eg. Bridging, doubling etc)

354 + 2 = 356
356 + 10 =366
366 + 90 = 1266
(use an appropriate method to add)

- I can determine the value of each digit in 3 digit numbers

1. Place value grid with counters and digits

2. Part part whole model



- I can line up 2 digit and 3 digit numbers in columns

E.g. 345 and 84

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest

3	4	5			8	4
3	4	5				
	8	4				

- I can add two 2 digit numbers using the formal written method of addition without crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits
6. Add the tens digits

6	3	+	2	6	=
	6	3			
+	2	6			
	8	9			

- I can add two 2 digit numbers using the formal written method of addition when crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ones column)
6. Add the tens digits (and any carried digits, making sure you cross it out to remember you have added it)

3	7	+	4	6	=
	3	7			
+	4	6			
	8	3			
	1				

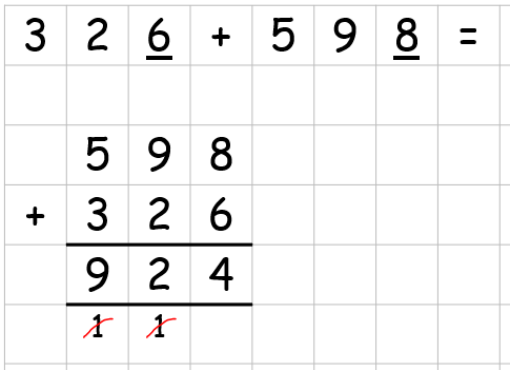
- I can add two 3 digit numbers using the formal written method of addition without crossing the boundary

3	4	6	+	5	4	2	=	
	3	4	6					
+	5	4	2					
	8	8	8					

1. Underline the ones
1. Write the biggest number first
2. Write the next number's ones digit
3. Fill in the rest
4. Add the ones digits
5. Add the tens digits
6. Add the hundreds

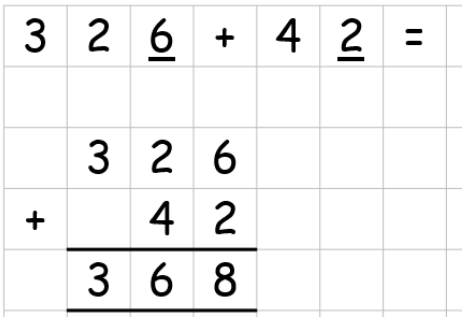
- I can add two 3 digit numbers using the formal written method of addition when crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ones column)
6. Add the tens digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the tens column)
7. Add the hundreds digits (and any carried digits, making sure you cross it out to remember you have added it)



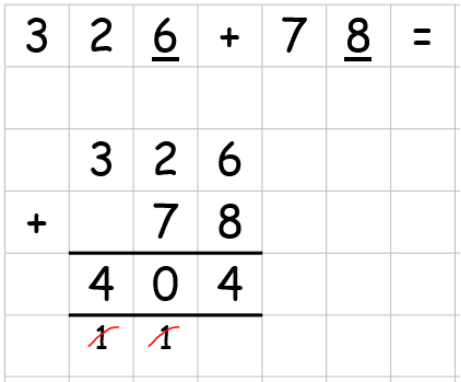
- I can add a 3 digit and 2 digit number using the formal written method of addition without crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits
6. Add the tens digits
7. Add the hundreds



- I can add a 3 digit and 2 digit number using the formal written method of addition when crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ones column)
6. Add the tens digits (and any carried digits, making sure you cross it out to remember



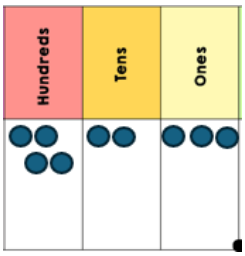
you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the tens column)

7. Add the hundreds digits (add any carried digits, making sure you cross it out to remember you have added it)

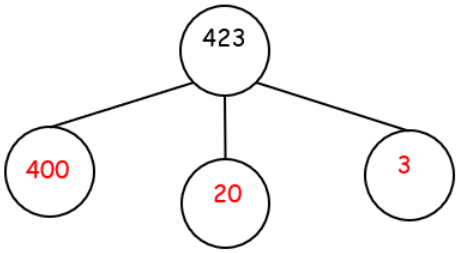
Year Four

- I can partition 4 digit numbers

1. Place value grid with counters and digits



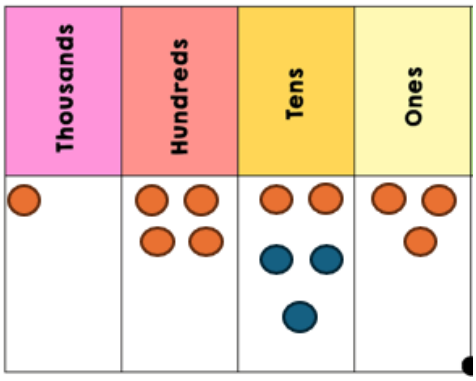
2. Part part whole model



- I can multiples of 10 to 3 digit numbers without crossing a boundary
  See previous year group
- I can add multiples of 10 to 3 digit numbers when crossing a boundary
  See previous year group
- I can add multiples of 100 without crossing a boundary
  See previous year group
- I can add multiples of 100 to 3 digit numbers when crossing a boundary
  See previous year group
- I can add multiples of 10 to 4 digit numbers without crossing a boundary

  E.g. 1423 + 30 =

1. Place value grid

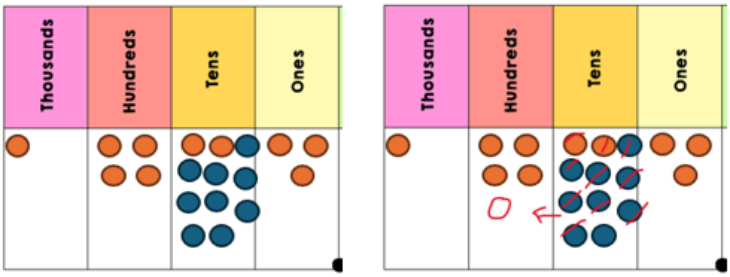


2. Rule:
    1. Underline the tens column (you are adding a multiple of 10)
    2. Add them together
    3. Everything else stays the same

$$1423 + 30 = 1453$$

- I can add multiples of 10 to 4 digit numbers when crossing a boundary

1. Place value grid



2. Rule:
    1. Underline the tens column (you are adding a multiple of 10)
    2. Add them together
    3. Exchange by giving the extra '10' to the hundreds

$$1423 + 90 = 1513$$

- I can add multiples of 100 to 4 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 100 to 4 digit numbers when crossing a boundary
  Use previous methods

- I can add multiples of 1,000 to 4 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 1,000 to 4 digit numbers when crossing a boundary

Use previous methods

- I can add two 2 digit numbers using known facts

  E.g. 20 + 30 = → 2 + 3 = 5 so 20 + 30 = 50

  1. Place value grid and counters

  2. Explain why we put the 0s back on (each number is ten times bigger so the answer needs to be ten times bigger)

- I can add two 3 digit numbers using known facts

  E.g. 200 + 300 = → 2 + 3 = 5 so 200 + 300 = 500

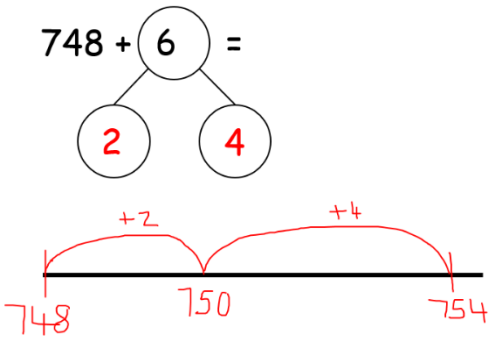
  1. Place value grid and counters

  2. Explain why we put the 0s back on (each number is one hundred times bigger so the answer needs to be ten one hundred bigger)

- I can add a 3 digit and 1 digit number using bridging

  E.g. 748 + 6 =

  1. Number line using part part whole model



- I can add a 4 digit and 1 digit number using bridging
  Use previous method

- I can add a 4 digit and 2 digit number using bridging
  Use previous method

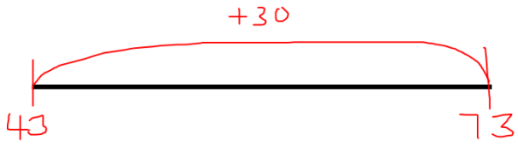
- I can add two 2 digit numbers using compensating

E.g. + _9 and + _1
GD → + _7 + _8 + _2 + _3 etc

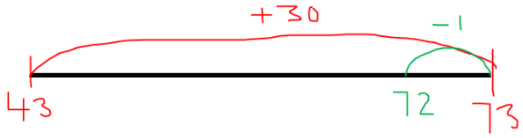
43 + 29 =

1. Tens frame

2. Bead string



3. Number line



- *First add 30*
- *Then subtract 1*

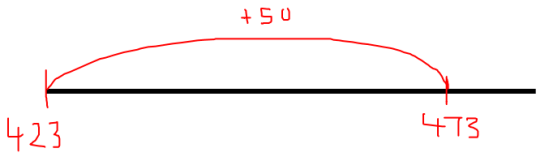
- I can add a 3 digit and 2 digit number using compensating

E.g. + _ _ 9 and + _ _1
GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

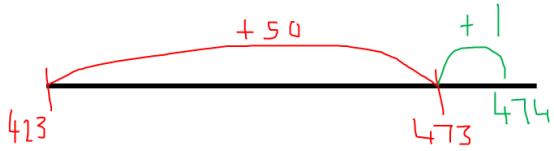
423 + 51 = 474

1. Tens frame

2. Bead string



3. Number line

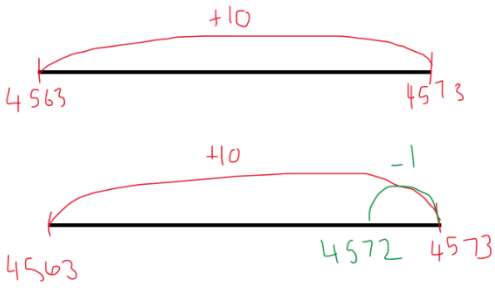


- *First add 50*
- *Then add 1*

- I can add a 4 digit and 1 digit number using compensating

  E.g. 4563 + 9 and 4532 + 1
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

4563 + 9 =



- I can add a 4 digit and 2 digit number using compensating
  Use previous method

  E.g. 4563 + 49 and 4532 + 31
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add a 4 digit and 3 digit number using compensating
  Use previous method

  E.g. 4563 + 409 and 4532 + 301
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add two 4 digit numbers using compensating
  Use previous method

  E.g. 4563 + 5009 and 4532 + 2001
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add two multiples of 10 using doubling

  E.g. 60 + 70 =

  2. Bar model

130		
60	70	
60	60	10

- I can add two multiples of 100 using doubling
  Use previous method

- I can add two multiples of 1000 using doubling
  Use previous method

- I can add three multiples of 10 using reordering using number bonds

E.g. 30 + 50 + 70 =

30 + 50 + 70 =

30 + 70 + 50 =

100 + 50 = 150

- I can add three multiples of 100 using reordering using number bonds

Use previous method

- I can add three multiples of 1000 using reordering using number bonds

Use previous method

- I can three 2 digit and 1 digit numbers using reordering

Looking for known facts

E.g. 23 + 14 + 17 =

23 + 14 + 17 =

23 + 17+ 14 =

40 + 14 = 54 (number bonds)

- I can add three numbers together (4 digit – 1 digit) using reordering

Use previous method

- I can add a 3 digit and 1 digit number using partitioning without crossing a boundary

3. Place value grid with counters and digits

354 + 3 =

4. Partitioning

354 + 3 =

300 50 4 3

300

50

4 + 3 = 7

300 + 50 + 7 = 357

- I can add a 3 digit and 2 digit number using partitioning without crossing a boundary

Use previous method

- I can add a 3 digit and 3 digit number using partitioning without crossing a boundary
  Use previous method

- I can add a 4 digit and 1 digit number using partitioning without crossing a boundary
  Use previous method

- I can add a 4 digit and 2 digit number using partitioning without crossing a boundary
  Use previous method

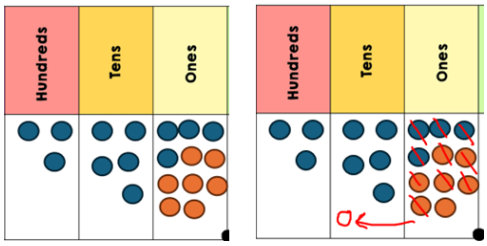
- I can add a 4 digit and 3 digit number using partitioning without crossing a boundary
  Use previous method

- I can add a 4 digit and 4 digit number using partitioning without crossing a boundary
  Use previous method

- I can add a 3 digit and 1 digit number using partitioning when crossing a boundary

  3. Place value chart and counters

     4. Put in the largest number
     5. Add the ones
     6. Make a 10 using the ones and put into the tens column



  4. Partitioning

     3. Start with the largest number
     4. Add the ones (may need to use another method to support eg. Bridging, doubling etc)

354 + 7 =

7

354 + 7 = 361

(use an appropriate method e.g. bridging)

- I can add a 3 digit and 2 digit number using partitioning when crossing a boundary
  Use previous method

- I can add a 3 digit and 3 digit number using partitioning when crossing a boundary
  Use previous method

- I can add a 4 digit and 1 digit number using partitioning when crossing a boundary

Use previous method

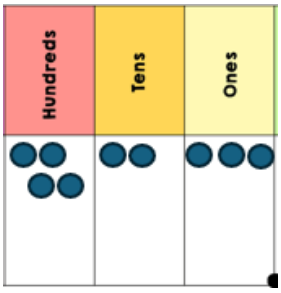
- I can add a 4 digit and 21 digit number using partitioning when crossing a boundary
  Use previous method

- I can add a 4 digit and 3 digit number using partitioning when crossing a boundary
  Use previous method

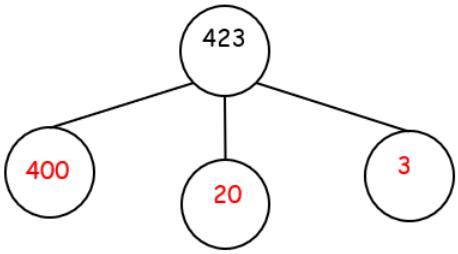
- I can add a 4 digit and 4 digit number using partitioning when crossing a boundary
  Use previous method

- I can determine the value of each digit in 4 digit numbers

  1. Place value grid with counters and digits



  2. Part part whole model



- I can line up 2 digit and 3 digit numbers in columns

  E.g. 345 and 84

  1. Underline the ones
  2. Write the biggest number first
  3. Write the next number's ones digit
  4. Fill in the rest

3	4	5			8	4
3	4	5				
	8	4				

- I can line up 2 digit, 3 digit and 4 digit numbers in columns
  Use previous method

- I can add two 2 digit numbers using the formal written method of addition without crossing the boundary

See Year Three

- I can add two 2 digit numbers using the formal written method of addition when crossing the boundary

  See Year Three

- I can add two 3 digit numbers using the formal written method of addition without crossing the boundary

  See Year Three

- I can add two 3 digit numbers using the formal written method of addition when crossing the boundary

  See Year Three

- I can add a 3 digit and 2 digit number using the formal written method of addition without crossing the boundary

  See Year Three

- I can add a 3 digit and 2 digit number using the formal written method of addition when crossing the boundary

  See Year Three

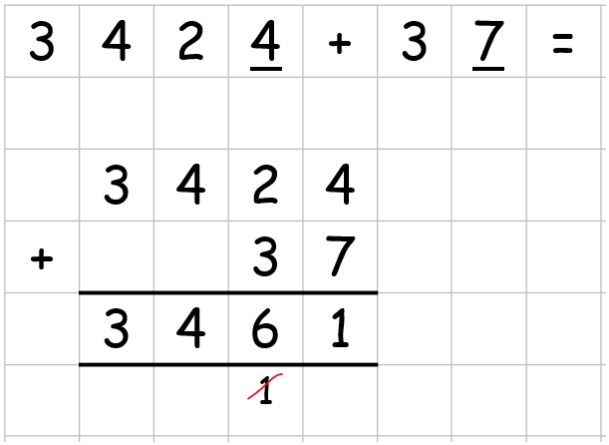
- I can add a 4 digit number and 2 digit number using the formal written method of addition without crossing the boundary

  1. Underline the ones
  2. Write the biggest number first
  3. Write the next number's ones digit
  4. Fill in the rest
  5. Add the ones digits
  6. Add the tens digits
  7. Add the hundreds
  8. Add the thousands

3	4	2	1	+	3	5	=
	3	4	2	1			
+			3	5			
	3	4	5	6			

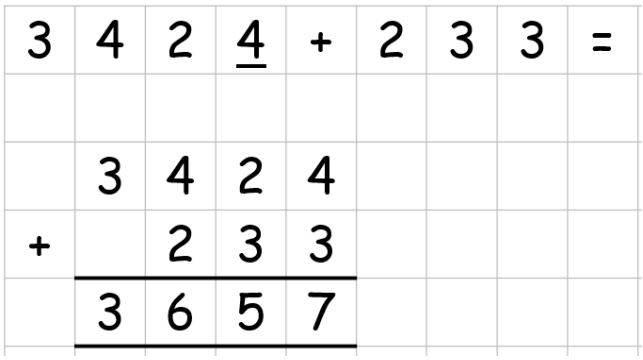
- I can add a 4-digit number and 2-digit number using the formal written method of addition when crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ones column)
6. Add the tens digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the tens column)
7. Add the hundreds digits (and any carried digits, making sure you cross it out to remember you have added it)
8. Add the thousands digits



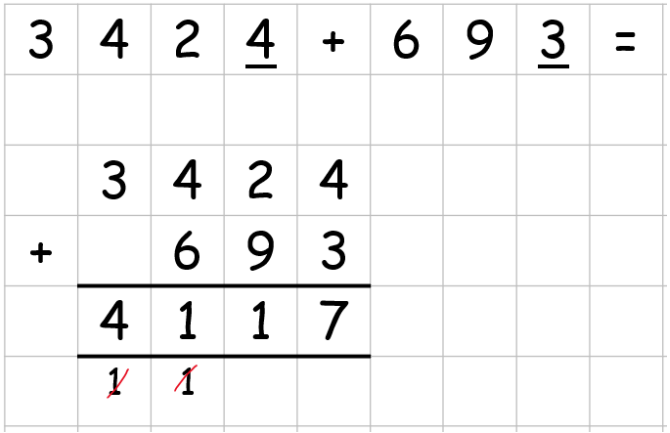
- I can add a 4 digit number and 3 digit number using the formal written method of addition without crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits
6. Add the tens digits
7. Add the hundreds
8. Add the thousands



- I can add a 4 digit number and 3 digit number using the formal written method of addition when crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ones column)
6. Add the tens digits (add any carried digits, making sure you



cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the hundreds column)

7. Add the hundreds digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the thousands column)
8. Add the thousands digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ten thousands column)

- I can add a 4 digit number and 4 digit number using the formal written method of addition without crossing the boundary

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits
6. Add the tens digits
7. Add the hundreds
8. Add the thousands

3	4	2	4	+	1	2	3	3	=
	3	4	2	4					
+	1	2	3	3					
	4	6	5	7					

- I can add a 4 digit number and 4 digit number using the formal written method of addition when crossing the boundary

1.

1.
Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits (if it is 10 or above, you need to carry the tens digit into the tens column and write the ones digit in the ones column)

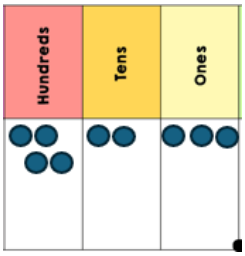
3	4	2	4	+	1	6	9	3	=
	3	4	2	4					
+	1	6	9	3					
	5	1	1	7					
	1	1							

6. Add the tens digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the hundreds column)
7. Add the hundreds digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the thousands column)
8. Add the thousands digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ten thousands column)

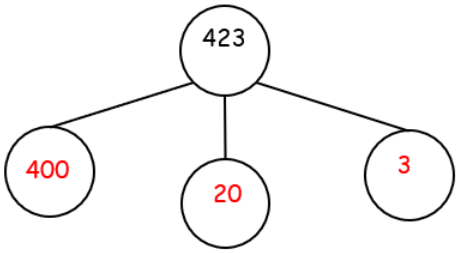
Year Five

- I can partition 5 and 6 digit numbers

1. Place value grid with counters and digits



2. Part part whole model



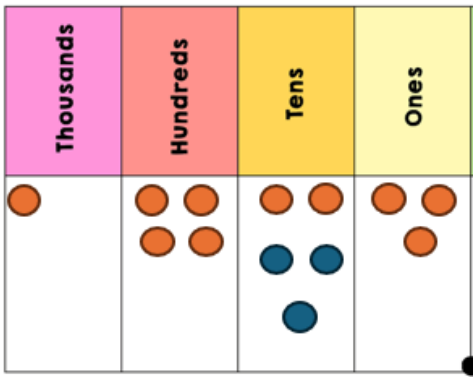
- I can partition numbers with 1 decimal place
  Use previous methods

- I can partition numbers with 2 decimal places
  Use previous methods
- I can add multiples of 10 to 4 digit numbers without crossing a boundary

  E.g. 1423 + 30 =

u

1. Place value grid

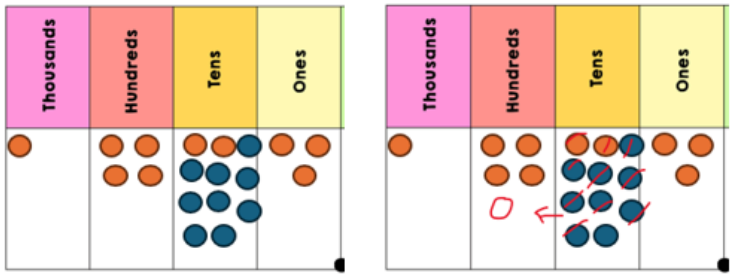


2. Rule:
    1. Underline the tens column (you are adding a multiple of 10)
    2. Add them together
    3. Everything else stays the same

$$1423 + 30 = 1453$$

- I can add multiples of 10 to 4 digit numbers when crossing a boundary

1. Place value grid



2. Rule:
    1. Underline the tens column (you are adding a multiple of 10)
    2. Add them together
    3. Exchange by giving the extra '10' to the hundreds

$$1423 + 90 = 1513$$

- I can add multiples of 100 to 4 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 100 to 4 digit numbers when crossing a boundary
  Use previous methods

- I can add multiples of 1,000 to 4 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 1,000 to 4 digit numbers when crossing a boundary
  Use previous methods

- I can add multiples of 10 to 5 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 100 to 5 digit numbers when crossing a boundary
  Use previous methods

- I can add multiples of 1,000 to 5 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 10,000 to 5 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 10 to 6 digit numbers when crossing a boundary
  Use previous methods

- I can add multiples of 100 to 6 digit numbers when crossing a boundary
  Use previous methods

- I can add multiples of 1,000 to 6 digit numbers when crossing a boundary
  Use previous methods

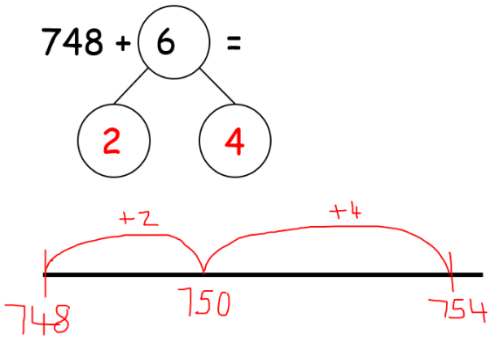
- I can add multiples of 10,000 to 6 digit numbers without crossing a boundary
  Use previous methods

- I can add multiples of 100,000 to 6 digit numbers without crossing a boundary
  Use previous methods

- I can add a 4 digit and 1 digit number using bridging

  E.g. 748 + 6 =

  1. Number line using part part whole model



- I can add a 4 digit and 2 digit number using bridging
  Use previous method

- I can add a 5 digit and 1 digit number using bridging

Use previous method

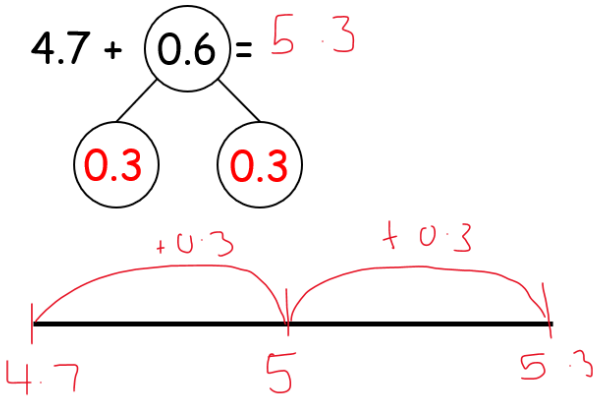
- I can add a 5 digit and 2 digit number using bridging
  Use previous method

- I can add a 6 digit and 1 digit number using bridging
  Use previous method

- I can add a 6 digit and 2 digit number using bridging
  Use previous method

- I can add two numbers with one decimal place using bridging
  E.g. 4.7 + 0.6 =

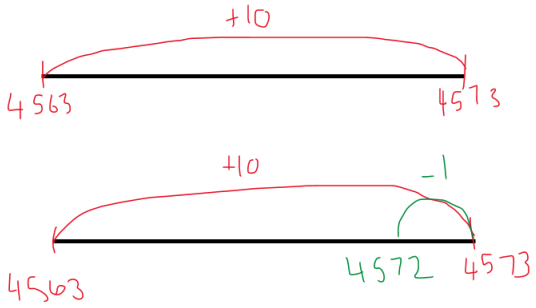
  1. Number line using part part whole model



- I can add a 4 digit and 1 digit number using compensating

  E.g. 4563 + 9 and 4532 + 1
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

4563 + 9 =



- I can add a 4 digit and 2 digit number using compensating
  Use previous method

  E.g. 4563 + 49 and 4532 + 31
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add a 4 digit and 3 digit number using compensating
  Use previous method

  E.g. 4563 + 409 and 4532 + 301
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add two 4 digit numbers using compensating
  Use previous method

  E.g. 4563 + 5009 and 4532 + 2001
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add a 5 digit number and 1 digit number using compensating (including decimals)
  Use previous method

  E.g. 44,563 + 9
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add a 5 digit number and 2 digit number using compensating (including decimals)
  Use previous method

  E.g. 44,563 + 29 and 43,422 + 69
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add a 5 digit number and 3 digit number using compensating (including decimals)
  Use previous method

  E.g. 44,563 + 209 and 43,422 + 609
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add a 5 digit number and 4 digit number using compensating (including decimals)
  Use previous method

  E.g. 44,563 + 2109 and 43,422 + 6109
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- I can add two 5 digit numbers using compensating (including decimals)
  Use previous method

  E.g. 44,563 + 40069 and 43,422 + 60109
  GD → + _ _ 7 + _ _ 8 + _ _ 2 + _ _ 3 etc

- Only when other strategies are secure (GD)

- I can add two multiples of 100 using doubling
  Use previous method

- I can add two multiples of 1000 using doubling
  Use previous method

- I can add two multiples of 10,000 using doubling
  Use previous method

- I can add two multiples of 100,000 using doubling
  Use previous method

- I can add two decimal numbers using doubling
  Use previous method

- I can add three multiples of 100 using reordering using number bonds
  Use previous method

- I can add three multiples of 1000 using reordering using number bonds
  Use previous method

- I can add three multiples of 10,000 using reordering using number bonds
  Use previous method

- I can add three multiples of 100,000 using reordering using number bonds
  Use previous method

- I can add three numbers together (4 digit – 1 digit) using reordering
  Use previous method

- I can add three numbers together (6 digit – 1 digit) using reordering
  Use previous method

- I can add a 4 digit and 1 digit number using partitioning without/when crossing a boundary
  Use previous method

- I can add a 4 digit and 2 digit number using partitioning without/when crossing a boundary
  Use previous method

- I can add a 4 digit and 3 digit number using partitioning without/when crossing a boundary
  Use previous method

- I can add a 4 digit and 4 digit number using partitioning without/when crossing a boundary
  Use previous method

- I can add a 5 digit and 1 – 5 digit numbers using partitioning without crossing a boundary
  Use previous method

- I can add a 5 digit and 1 – 5 digit numbers using partitioning when crossing a boundary
  Use previous method

- I can add a 6 digit and 1 – 5 digit numbers using partitioning without crossing a boundary
  Use previous method

- I can add a 6 digit and 1 – 5 digit numbers using partitioning when crossing a boundary
  Use previous method

- I can line up 2 digit, 3 digit and 4 digit numbers in columns
  Use previous method

- I can line up 5 and 6 digit numbers in columns (including decimals)
  Use previous method

- I can add a 4 digit number and 2 digit number using the formal written method of addition without crossing the boundary

- I can add a 4 digit number and 2 digit number using the formal written method of addition when crossing the boundary

- I can add a 4 digit number and 3 digit number using the formal written method of addition without crossing the boundary

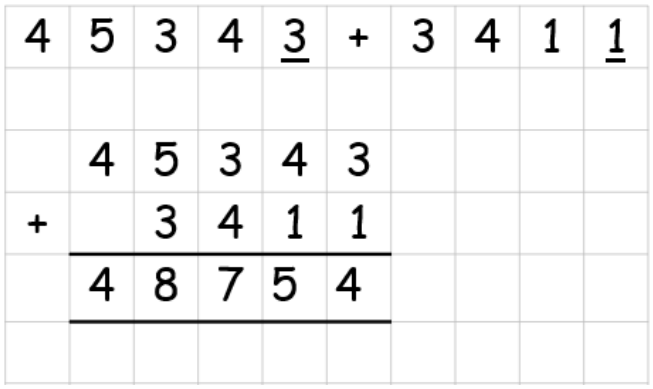
- I can add a 4 digit number and 3 digit number using the formal written method of addition when crossing the boundary

- I can add a 4 digit number and 4 digit number using the formal written method of addition without crossing the boundary

- I can add a 4 digit number and 4 digit number using the formal written method of addition when crossing the boundary

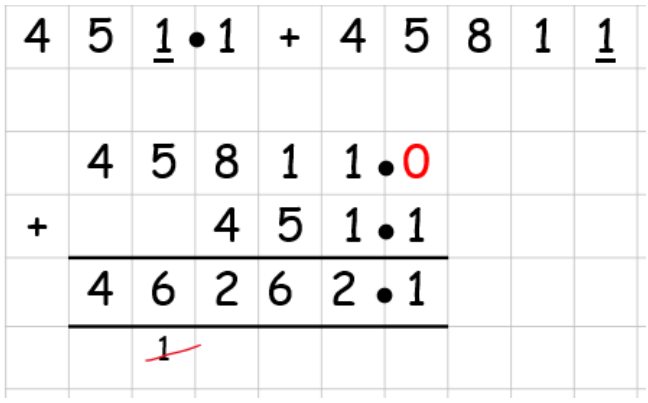
- I can add a 5 digit number and a 2, 3, 4 and 5 digit number using the formal written method of addition without crossing the boundary (including decimals)

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add the ones digits
6. Add the tens digits
7. Add the hundreds digits
8. Add the thousands digits
9. Add the ten thousands digits



- I can add a 5 digit number and a 2, 3, 4 and 5 digit number using the formal written method of addition when crossing the boundary (including decimals)

1. Underline the ones
2. Write the biggest number first
3. Write the next number's ones digit
4. Fill in the rest
5. Add any place holders
6. Add the tenths digits (if it is 10 or above, you need to carry the tens digit and write the ones digit in the tenths column)
7. Add the ones digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the tens column)
8. Add the tens digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the hundreds column)
9. Add the hundreds digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the thousands column)



10. Add the thousands digits (add any carried digits, making sure you cross it out to remember you have added it) (if it is 10 or above, you need to carry the tens digit and write the ones digit in the ten thousands column)

- I can add a 6 digit number and a 2, 3, 4 and 5 digit number using the formal written method of addition without crossing the boundary (including decimals)
  Use previous method

- I can add a 6 digit number and a 2, 3, 4 and 5 digit number using the formal written method of addition when crossing the boundary (including decimals)
  Use previous method

Year Six

Same as Year Five but include numbers beyond 1,000,000 for all objectives.