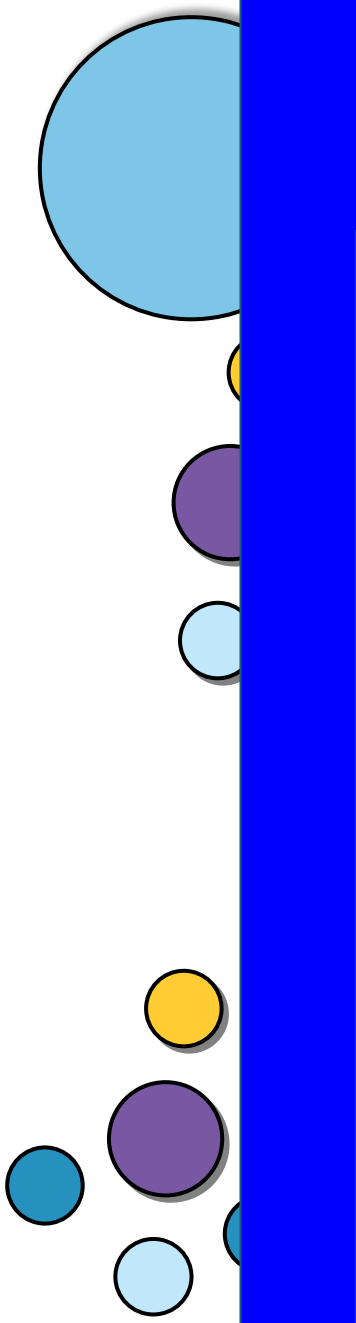
Java for Beginners

Level 4b

Mr. Teasdale

Levels of Java coding

- 1: Syntax, laws, variables, output
- 2: Input, calculations, String manipulation
- 3: Selection (IF-ELSE)
- **4: Iteration/Loops (FOR/WHILE)**
- 5: Complex algorithms
- 6: Arrays
- 7: File management
- 8: Methods
- 9: Objects and classes
- 10: Graphical user interface elements



What do you learn last time?

A simple variable (usually an int) that allows the loop to 'step through' Normally called "i"

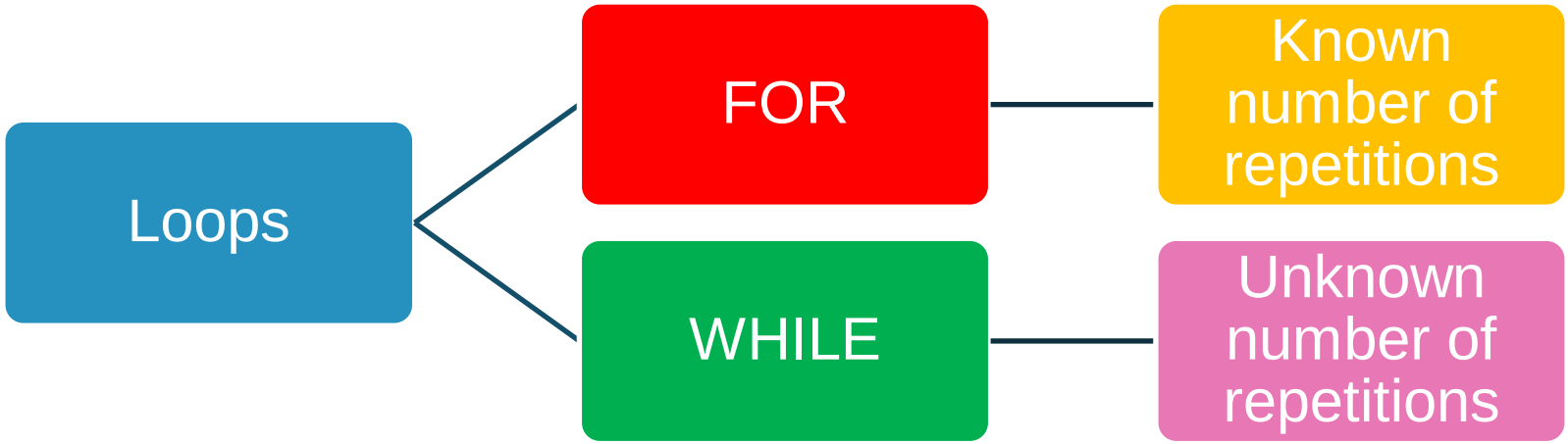
A **logic condition** (like in IF) that has to be **true** for the loop to continue.
Something Operator Something (usually involving the counter variable)

```
for(counter; condition; change)
```

What should happen to the **counter variable value** at the **end** of the loop
Usually **i++** or **i--**

What is the difference?

"Now count from one to ten..."



"Are we there yet?..."

A **variable** is created and **given a value** so that the loop will run. Usually **String**, **int** or **boolean**

A **logic condition** (like in IF) that has to be **true** for the loop to continue.
Something Operator Something (usually involving the counter variable)

```
condition variable
while(condition)
    change happens inside loop
```

There must be an **opportunity** for the condition variable to **change**, sometimes done in an **IF block**

Typical example (*while loop*)

Create int called **i**
Set **i** to **0**

Continue while **i** is **less than 3**

```
int i = 0;
while (i < 3)
{
   System.out.println(“gum”);
   i++;
}
```

At end of for loop, **increase i by 1**

Output

```
gum
gum
gum
```

Another example (*while loop*)

```
boolean done = false;
while (done == false)
{
        System.out.println("Are you done? Y/N ");
        String answer = kb.nextLine();
        if (answer.equals("Y") )
        {
                done = true;
        }
        System.out.println("Goodbye!");
}
```

Create boolean called done. **Set to false**

Continue while **done is false**

If user enters Y, **set done to true**

Output

```
Are you done? Y/N
Y
Goodbye!
```

Students struggle with…

FOR LOOPS – WHILE not so much

Some students will bizarely understand while loops much easier than for loops.

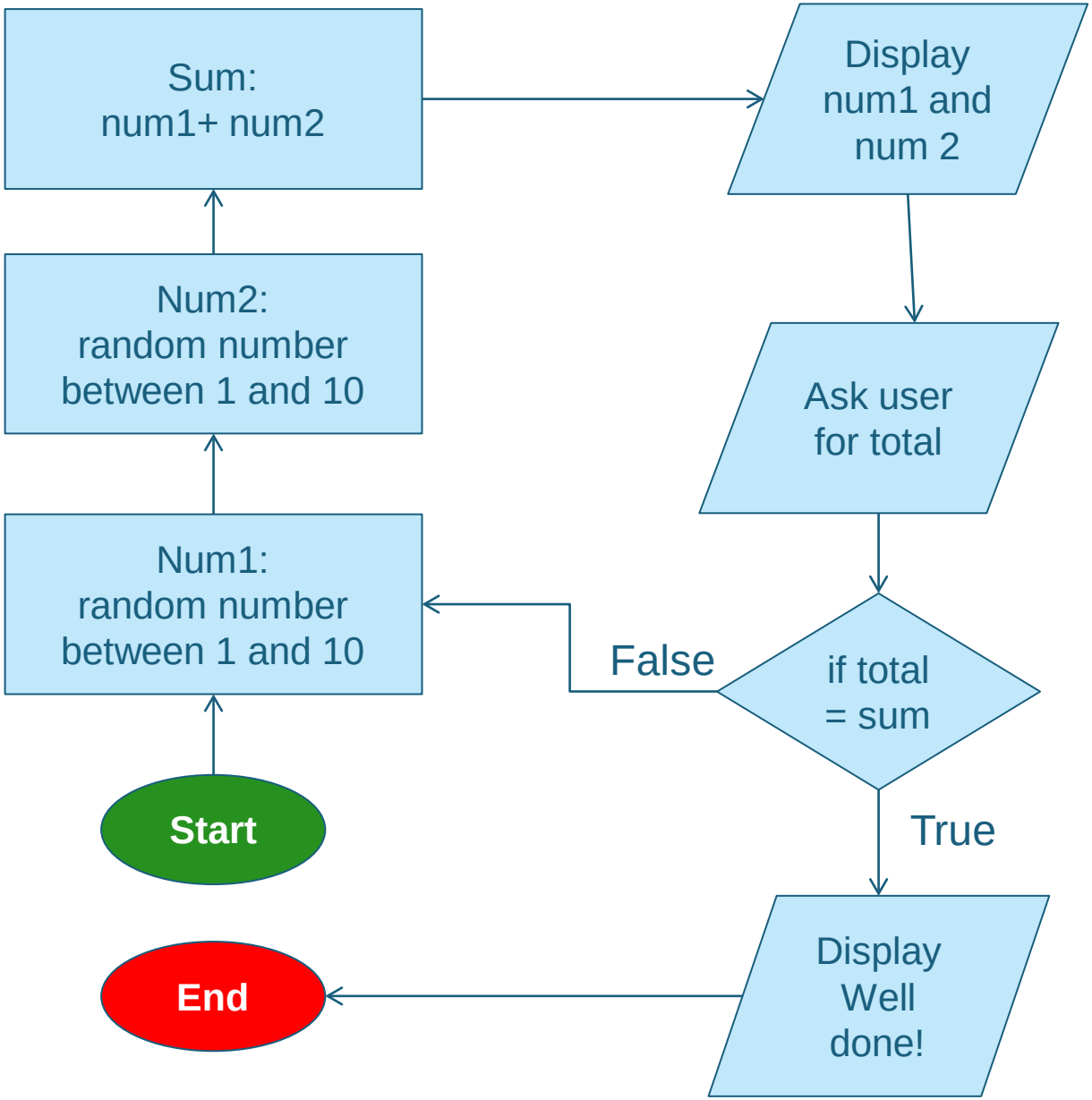
They can use whichever on they prefer

Most common mistake:

```
while(done == false);
```

Practice time

- ✓ Generate two random numbers between 1 and 10.
- ✓ Calculate their sum (num1 + num2).
- ✓ Ask the user for their sum.
- ✓ If they get it right, the program congratulates them and ends.
- ✓ If they get it wrong, the program repeats by generating two more numbers and asking the user again…


Sum:
num1+ num2
Display
num1 and
num 2
Num2:
random number
between 1 and 10
Ask user
for total
Num1:
random number
between 1 and 10
False
if total
= sum
Start
True
End
Display
Well
done!

Possible solution

```
boolean done = false;
while (done == false)
{
  int num1 = 1 +(int)(Math.random()*((10 - 1) + 1));
  int num2 = 1 +(int)(Math.random()*((10 - 1) + 1));
  int sum = num1 + num2;
  System.out.println("What is "+num1+" + "+num2+" ?");
  String answer = kb.nextLine();
  int total = Integer.parseInt(answer);
  if (total == sum)
  {
      done = true;
  }
}
System.out.println("Well done!");
```