Name: _____

Java While Loops

What are While Loops?

A **while loop** repeats a block of code over and over again as long as a condition is true. It's like telling the computer "keep doing this until something changes." When the condition becomes false, the loop stops.

While loops are perfect when you don't know exactly how many times you need to repeat something, but you know when to stop.

☐♀ Simple Analogy: While Loops are Like Running Laps

Imagine your coach says: **"Keep running laps while you still have energy"**

- **The condition:** "while you still have energy" (true or false)
- **The action:** "run one lap"
- **What happens:** You check if you have energy → If yes, run a lap → Check again → If yes, run another lap → Keep going until you're tired
- **When it stops:** When you don't have energy anymore (condition becomes false)

While Loop Process:
Check condition → Is it true? → Yes: Do action, go back to check → No: Stop the loop

Basic Pattern

```
while (condition) {
  // action to repeat
  // don't forget to change the condition!
}
```

Example 1: Counting from 1 to 5

```
int count = 1;

while (count <= 5) {
  System.out.println("Count: " + count);
  count++; // This changes count to count + 1
}

System.out.println("Done counting!");
```

What happens:

Output: Count: 1, Count: 2, Count: 3, Count: 4, Count: 5, Done counting!

Why it works: Each time through the loop, count increases by 1. When count becomes 6, the condition (count <= 5) becomes false, so the loop stops.

Example 2: Asking for the right password

```java
Scanner input = new Scanner(System.in);
String password = "";

while (!password.equals("secret")) {
  System.out.print("Enter password: ");
 password = input.nextLine();
}

System.out.println("Access granted!");
```

What happens: The program keeps asking for a password until the user types "secret".

Example 3: Adding numbers until we reach 100

```java
int sum = 0;
int number = 1;

while (sum < 100) {
 sum = sum + number;
 number++;
  System.out.println("Added " + (number-1) + ", sum is now: " + sum);
}

System.out.println("Final sum: " + sum);
```

What happens: We keep adding 1, then 2, then 3, etc., until the sum reaches or passes 100.

⚠ DANGER: Infinite Loops!

Always make sure your condition can become false!

```java
// BAD - This will never stop! int x = 1;
while (x > 0) {
  System.out.println("This runs forever!");
// x never changes, so x > 0 is always true!
}
```

Always include something inside the loop that changes the condition or include a `break`!

Common While Loop Patterns

Pattern 1: Counter Loop

```java
int i = 0;
while (i < 10) {
 // do something
 i++; // increment counter
}
```

Pattern 2: User Input Loop

```
String answer = "";
while (!answer.equals("quit")) {
 // get user input
 answer = input.nextLine();
}
```

Pattern 3: Accumulator Loop

```
int sum = 0;
int i = 1;
while (i <= 100) {
 sum += i; // add i to sum
 i++;
}
```

Important Rules to Remember

- **Always change the condition** inside the loop, or it will run forever!
- The condition is checked **before** each time the loop runs
- If the condition starts as false, a while loop won't run at all
- Common operators: **++** (add 1), **--** (subtract 1), **+=** (add to variable)
- Use meaningful variable names like **`count`**, **`sum`**, **`answer`**

Your Turn: Write Your Own Definition

What is a while loop? How would you explain it to a friend?

Write your definition in your own words:

Think of a real-life situation where you repeat an action until something changes. Describe it:

Example: "I keep checking my phone while I'm waiting for a text message."

Your example:

Why is it important to change the condition inside the loop?