QUIZ

In this method definition, what does `miles` represent?

```
public void run(int miles) {
}
```

A Java specific keyword

A constructor

An object belonging to the `Miles` class

A parameter of type `int`

👏 Yes! It is the `int` input to the `run()` method.

Fill in the code that will call the `addSugar()` method with 7 as an input.

```
public class Coffee {

  public Coffee(){

  }

  public void addSugar(int sugarCubes){
    System.out.println("Added " + sugarCubes + " sugar cubes!");
  }

  public static void main(String[] args){
    Coffee myCup = new Coffee()
      myCup.addSugar(7);

  }

}
```

👏 You got it!

When defining a method, the `void` keyword specifies which of the following?

The method does not take any parameters

The method can run without being called

The method returns default Java errors

The method does not return a value

👏 Yes! `void` means that nothing is returned from the method call.

Which method signature would make a method that returns an `int` and takes in a `String`?

```
public void countLetters(String message, int numLetters)
```

```
public int countLetters(String message)
```

👏 Yes! The return type is `int` and the input is of type `String`.

```
public intReturn()
```

```
public String countLetters(int numLetters)
```

Is there an error that will prevent this code from running?

```java
class Coffee{

  public Coffee(){
  }

  public void stir(){
    String stirring = "Stirring the coffee!";
  }

  public static void main(String[] args){
    Coffee myCup = new Coffee();
    myCup.stir();
    System.out.println(stirring);
  }

}
```

Yes! The variable `stirring` cannot be accessed in the `main()` method.

👏 This is true. `stirring` only exists within the `stir()` method.

An object named `dalmatian` belongs to the Spot class. Which of the following correctly calls the `run` method on `dalmatian`?

Spot[run]

Spot.run();

dalmatian.run();

👏 Yes! The object is `dalmatian`, and the method `run()` is being called on it.

Spot.dalmatian()

Which line of code is **NOT** an example of calling a method?

```
"hello".equals("goodbye");
```

```
System.out.println("More sugar please!");
```

```
myCoffeeCup.emptyCup();
```

```
int sugarCubes = 7;
```

👏 Correct! This line of code represents a variable being declared and initialized, not a method being called.