What questions did you have from the video last night?

Students, write your response!

Day 1

passing values
writing void methods

Drag the colored dot that matches the color of the definition to the appropriate word. Some words will not be used!

A set of statements grouped together to form an operation.

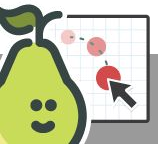
To use a method in a program, you _____ it.

The local variable in a method header that stores the data passed into the method.

A value or variable in a method call that stores the data being passed into the method.

A contract written that specifies the conditions that must be true in order for the method to behave as expected

1) precondition
2) postcondition
3) parameter
4) argument
5) call
6) write
7) method
8) command



Students, drag the icons!

Answers

Slide 2:

Method: A set of statements grouped together to form an operation.

Call: To use a method in a program, you call it.

Parameter: The local variable in a method header that stores the data passed into the method.

Argument: A value or variable in a method call that stores the data being passed into the method.

Precondition: A contract written that specifies the conditions that must be true in order for the method to behave as expected

Postcondition: statement that must be true after the execution of a method, describes outcome and return value

Which of the following is the correct syntax for a method header with parameters?

a. public static void example(int x,y) {
b. public static (int x, int y) example() {
c. public static void example(x, y) {
d. public static void example(x: int, y: int) {
e. public static void example(int x, int y) {

Answer

E

Which of the following is NOT a legal call to the method:

```
public static void powerOfTwo(double x)  {
    System.out.println(Math.pow(x, 2));
}
```

(A)
```
powerOfTwo(4);
```

(B)
```
double a = 89.003;
powerOfTwo(a - .003);
```

(C)
```
powerOfTwo(0.5);
```

(D)
```
powerOfTwo("n");
```

(E) All of the above are legal method calls.

Answer

D

Which of the following is a legal call to the method with the header:

```
public static void guess(double x, int y, String z)  {
```

(A) `guess(4, 3, "cookies");`

(B) `guess(3, 4.0, "cookies");`

(C) `guess(3.0, 4.0, "cookies");`

(D) `guess("cookies", 3, 4);`

(E) `guess(4.0, 3, cookies);`

Students choose an option

Answer

A

Let’s discuss!

What’s wrong with the following code? Where do you see errors?

```
  public static void  f(double   a);
{
         double a;
         System.out.println(a) ;
}
```

Answers

can’t redeclare a variable with same name in method,

there is an ; after the header

no pre and postconditions

What is printed as a result of running the main method? Make a table to help you keep track of values!

```
public class MysterySoda
{
    public static void main(String[] args) {
        String soda = "coke";
        String pop = "pepsi";
        String pepsi = "soda";
         carbonated(soda, pop, pepsi);
    }


    public static void carbonated(String coke, String soda, String pop)
    {
     System.out.println("say " + soda + " not " + pop + " or " + coke);
    }
}
```

variable	value

output:

Students, draw anywhere on this slide!

Answer

`say pepsi not soda or coke`

variable	value
soda	“coke”
pop	“pepsi”
pepsi	“soda”

```
carbonated(soda, pop, pepsi)--> carbonated(“coke”, “pepsi”, “soda)


                                “coke”       “pepsi”       “soda”
public static void carbonated(String coke, String soda, String pop)
```

What is the output from running the following program. BE CAREFUL!!!!!!!

```
public class Swapping
{
    public static void main(String[] args) {
        int  x = 12;
        int  y = 30;
        swap (x, y);
        System.out.println ("X: " + x  + "\nY: " + y);
    }


    public static void swap (int  x, int y)
    {
        int  temp = x;
        x = y;
        y = temp;
    }
}
```

Students, write your response!

Answer

```
X: 12

Y: 30
```

What output will be produced by this program?

```
public class Mystery {
     public static void main(String[ ] args) {
          int a = 6;
          int b = 3;
          strangeMethod(a,b);
          System.out.print(a + " " + b);
     }

     public static void strangeMethod(int a, int b) {
          a += b;
          b *= a;
          System.out.print(a + " " + b + " ");
     }
}
```

Students choose an option

Answer

E

Write a method `sumThree` that has 3 integer parameters and prints their sum. Don't forget pre and postcondition!

Then, write a valid call to the method.

Answers

```
/* precondition: must be called with three int variables/values
postcondition: will print to the console the sum of the three arguments */
public static void sumThree(int a, int b, int c) {

    System.out.print(a+b+c);

}

public static void main(String[]  args) {

    sumThree(1,2,3); //methods can be called in main method

}
```

Write the method `printGrade` that accepts 2 arguments, a person's name and a test score, and prints a message (see below for example). Don't forget pre and postcondition!

Then, write the call to the method that would print the line to the console.

```
Harry's test score is 84.5
```

Answers

/* precondition: must be called with a String representing Student name and double representing Student's test score, testScore >= 0
postcondition: will print to the console line with student's name and test score*/

```
public static void printGrade(String name, double testScore) {

    System.out.println(name + "'s test score is " + testScore);

}

printGrade("Harry", 84.5);
```

Write a method that accepts a number of seconds and prints the correct number of hours, minutes and seconds. This method must be called `realTime` and its parameter must be an integer. Don't forget pre and postcondition.

When you call the method with the argument 6342, it would print the following indented lines (with tabs!):

```
Hours: 1
Minutes: 45
Seconds: 42
```

Answer

/* precondition: must be called with int variable representing a number of seconds, greater than or equal to 0
postcondition: will print to the console the whole number of hours, minutes and seconds in the number of seconds passed in */

```
public static void realTime(int numSeconds) {
        int hours = numSeconds/3600;
        numSeconds %= 3600;
        int minutes = numSeconds/60;
        numSeconds %= 60;
        System.out.println("\tHours: " + hours);
        System.out.println("\tMinutes: " + minutes);
        System.out.println("\tSeconds: " + numSeconds);
}
```

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Write a method called `distance` that has four integer parameters x1, y1, x2, and y2 representing coordinates and prints the distance between points (x1, y1) and (x2, y2) on the Cartesian plane. Don't forget pre and postcondition.

For example, the call of `distance(1, 0, 4, 4)` would print:

The distance between (1,0) and (4,4) is: 5.0

Answers

/* precondition: must be called with four int variables, first two represent x and y coordinates of one point and last two represent x and y coordinates of second point
postcondition: will print to the console the distance between the two point */

```
public static void distance(int x1, int y1, int x2, int y2) {
        double dstance =
        Math.sqrt(Math.pow(x1-x2,2)+ Math.pow(y1-y2,2));
        System.out.println("The distance between (" + x1 + "," + y1 +
        ") and (" + x2 + "," + y2 + ") is: " + dstance);
}
```

The following program produces the output at left, but it has poor structure and redundancy. Rewrite the class by adding *at least* **two static methods to reduce the redundancy of the program**. Then rewrite the main method with these calls.

```
1   public class FightSong {
 2    public static void main(String[] args) {

 3       System.out.println("Go, team, go!");
 4       System.out.println("You can do it.");
 5       System.out.println();
 6       System.out.println("Go, team, go!");
 7       System.out.println("You can do it.");
 8       System.out.println("You're the best,");
 9       System.out.println("In the West.");
10       System.out.println("Go, team, go!");
11       System.out.println("You can do it.");
12       System.out.println();
13       System.out.println("Go, team, go!");
14       System.out.println("You can do it.");
15       System.out.println("You're the best,");
16       System.out.println("in the West.");
17       System.out.println("Go, team, go!");
18       System.out.println("You can do it.");
19       System.out.println();
20       System.out.println("Go, team, go!");
21       System.out.println("You can do it.");
22    }  // end of main

23 }  // end of FightSong
```

Answers

```
public static void encourageTeam() {
     System.out.println("Go, team, go!");
     System.out.println("You can do it.");
}

public static void stateBoth() {
     encourageTeam();
     System.out.println("You're the best,");
     System.out.println("In the West.");
}
```

```
public static void main(String[] args)
{

     encourageTeam();
     System.out.println();
     stateBoth();
     encourageTeam();
     System.out.println();
     stateBoth();
     encourageTeam();
     System.out.println();
     encourageTeam();
}
```