Methods

A method is a collection of statements that perform some specific task and return the result to the caller. A method can perform some specific task without returning anything. Methods allow us to reuse the code without retyping the code. In Java, every method must be part of some class which is different from languages like C, C++, and Python. Methods are time savers and help us to reuse the code without retyping the code. In Java, every method must be part of some class which is different from languages like C, C++, and Python.

```
public class MethodExample {
    public static void main(String[] args) {
    int a = 10;
    int b = 20;
    int c = sum(a, b);
    System.out.println("The sum of a and b is " + c);
    }

    public static int sum(int num1, int num2) {
    int result;
    result = num1 + num2;
    return result;
    }
}
```

A method can be returning nothing. In such case, the return type of the method is void.

```
public class MethodExample {
    public static void main(String[] args) {
    int a = 10;
    int b = 20;
    sum(a, b);
    }

    public static void sum(int num1, int num2) {
    int result;
    result = num1 + num2;
    System.out.println("The sum of a and b is " + result);
    }
}
```

Extra but IMPORTANT: In any Java project, there sould be one and only one public class. The name of the public class should be the same as the name of the file. For example, if the name of the file is `MethodExample.java`, the name of the public class should be `MethodExample`.

tldr: file name = class name

for class with `public static void main(String[] args)`

1. Fill the blank to print `time savor functions` in the output.

```
public class Punchcard {
    public static void printMessage() {
        System.out.println("time savor functions");
    }
    public static void main(String[] args) {
        ---------------
    }
}
```

A) `void(printMessage);`
B) `printVoid();`
C) `void();`
D) `printMessage;`
E) `printMessage();`

answer: E

---

2. What is wrong with the following code?

```
public class Punchcard {
    public static void main(String[] args) {
        int n = 5;
        System.out.println(n + " squared is " + toTwo(n));
        System.out.println(n + " cubed is " + toThree(n));
    }
}

class power {
    public static int toTwo(int n) {
        return n * n;
    }
    public static int toThree(int n) {
        return n * n * n;
    }
}
```

A) The class name should be `Power`
B) To call a method from another class, you need to use the class name
C) The variable `n` should be declared in the `toTwo` and `toThree` methods
D) The methods `toTwo` and `toThree` should be declared as `void`
E) It's not possible to have two classes in the same java file

answer: B

---

3. Fill the blank to print `Imagine 2 dragons` in the output.

```
public class Punchcard {
    public static ______ printMessage(_____) {
        return "Imagine " + x +  " dragons";
    }
    public static void main(String[] args) {
        System.out.println(printMessage(2));
    }
}
```

A) `void` and `int`
B) `int` and `int x`
C) `int` and `String x`

D) `String x` and `int`
E) `String` and `int x`

answer: E

---

4. what is the output of the following code?

```
public class Punchcard {
    public static void main(String[] args) {
        System.out.println(middle(1, 2, 3));
        System.out.println(middle(1, 3, 2));
    }
    public static int middle(int a, int b, int c) {
        if (a > b) {
            if (b > c) {
                return b;
            } else if (a > c) {
                return c;
            } else {
                return a;
            }
        } else {
            if (a > c) {
                return a;
            } else if (b > c) {
                return c;
            } else {
                return b;
            }
        }
    }
}
```

A) 2 and 2
B) 2 and 3
C) 3 and 2
D) error, It's not possible to call a methods two times in the same program
E) error, the method `middle` should be declared as `void`

answer: A

---

5. Why the following code fails to execute properly?

Extra: To call a method from another class, The method should be declared as `static`

```
public class Punchcard {
    public static void main(String[] args) {
        cat.meow;
    }
}
class cat {
    public static void eat() {
        System.out.println("kitty cat eating");
    }
    public static void meow() {
        System.out.println("meow");
    }
}
```

A) It's not possible to call a method from another class

B) The method `meow` should be declared as `String`
C) parameter state is missing, there need to be a `()` after the method name (`cat.meow()`)
D) The class cat should be written before the main method
E) The unused method `eat` should be removed

answer: C

---

6. What change will correct the code to print the sum of two integers?

```
public class punchcard {
    public static void main(String[] args) {
        int a = 5;
        int b = 10;
        System.out.println("The sum is: " + add(a, b));
    }

    public static int add(int x, int y) {
        return x - y;
    }
}
```

A) Change `return x - y;` to `return x + y;`
B) Change `System.out.println("The sum is: " + add(a, b));` to `System.out.println("The sum is: " + subtract(a, b))`
C) Remove the method `add`
D) Change `int a = 5;` to `int a = 15;`
E) Add a new method `public static int subtract(int x, int y) { return x + y; }`

answer: A

---

7. What modification will make the following code print "Hello, Java!"?

```
public class punchcard {
    public static void main(String[] args) {
        greet();
    }

    public static void greet(String message) {
        System.out.println(message);
    }
}
```

A) Add `String message = "Hello, Java!";` before `greet();`
B) Change `greet();` to `greet("Hello, Java!");`
C) Remove `String[] args` from `main` method
D) Change `public static void greet(String message)` to `public static void greet()`
E) Rename `greet(String message)` to `greet()`

answer: B

---

8. What modification will make the code properly compile and execute?

```
public class punchcard {
    public static void main(String[] args) {
        System.out.println(quote("Java is fun."));
    }
}

public class Text {
    public static String quote(String text) {
```

```
        return "Quote: " + text;
    }
}
```

A) Change `System.out.println(quote("Java is fun."));` to `System.out.println(Text.quote("Java is fun."));`
B) Remove `String[] args` from `main` method
C) Add `String quote = quote("Java is fun.");` before `System.out.println(quote);`
D) Remove the method `quote(String text)`
E) Change `return "Quote: " + text;` to `System.out.println("Quote: " + text);`

answer: A

---

9. What modification will fix the code to print "5"?

```
public class punchcard {
    public static void main(String[] args) {
        int result = square(2);
        System.out.println(result);
    }

    public static void square(int num) {
        return num * num + 1;
    }
}
```

A) Change `public static void square(int num)` to `public static int square(int num)`
B) Remove `String[] args` from `main` method
C) Add `return` before `num * num;` in the `square` method
D) Change `int result = square(2);` to `int result = square(5);`
E) Add `int num = 5;` before `int result = square(2);`

answer: A

---

10. What modification will fix the code to print "Hello, World!"?

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println(greet());
    }

    public static void greet() {
        return "Hello, World!";
    }
}
```

A) Change `System.out.println(greet());` to `System.out.println(HelloWorld.greet());`
B) Remove `String[] args` from `main` method
C) Change `public static void greet()` to `public static String greet()`
D) Change `public static void greet()` to `public static String greet() { return "Hello, World!"; }`
E) Remove the method `greet()`

answer: C

---

11. What change will make the code correctly print "true"?

```
public class Logic {
    public static void main(String[] args) {
        System.out.println(isEven(4));
    }
```

```
    public static boolean isEven(int number) {
        if (number % 2 == 0) {
            return "true";
        } else {
            return "false";
        }
    }
}
```

A) Change `public static boolean isEven(int number)` to `public static String isEven(int number)`
B) Remove `String[] args` from `main` method
C) Remove the quotes (") in `return "true";` and `return "false";` in the `isEven` method
D) Change `System.out.println(isEven(4));` to `System.out.println(Logic.isEven(4));`
E) Change `if (number % 2 == 0)` to `if (number % 2 != 0)`

answer: C

---

12. What change will fix the code to print "5"?

```
public class Numbers {
    public static void main(String[] args) {
        int result = add(2, 3);
        System.out.println(result);
    }

    public static int add(int a, int b) {
        a + b;
    }
}
```

A) Change `a + b;` to `return a + b;`
B) Remove `String[] args` from `main` method
C) Add `return` before `a + b;` in the `add` method
D) Change `int result = add(2, 3);` to `int result = add(5, 5);`
E) Add `int a = 5;` and `int b = 0;` before `int result = add(2, 3);`

answer: C

---

13. What change will make the code correctly print "10"?

```
public class Numbers {
    public static void main(String[] args) {
        int result = multiply(2, 5);
        System.out.println(result);
    }

    public static int multiply(int x, int y) {
        return x + y;
    }
}
```

A) Change `return x + y;` to `return x * y;`
B) Remove `String[] args` from `main` method
C) Change `int result = multiply(2, 5);` to `int result = multiply(5, 2);`
D) Add `int x = 5;` and `int y = 5;` before `int result = multiply(2, 5);`
E) Change `public static int multiply(int x, int y)` to `public static void multiply(int x, int y)`

answer: A

---

14. Fill the blank so the fox talks.

```
public class Punchcard {
    public static void main(String[] args) {
        System.out.println("What does the fox say?");
        fox.talk();

    }
}
class ___ {
    public static ____ talk() {
        ___________________"Screech"_;
    }
}
```

Certainly! Here are five choices for completing the code:

A) `fox`, `void`, `return`
B) `fox`, `String`, `System.out.println()`
C) `fox`, `void`, `System.out.println()`
D) `Animal`, `void`, `return`
E) `Fox`, `void`, `System.out.print()`

answer: C