

Year 11 Computer Science

Topic 1 - Input/Output, Data Types, and Variables

Tutorial 1

When we wish to output a value to the console, we use the statement:

System.out.**print**(insertValueThatYouWishToPrintHere) to print on the same line.

System.out.**println**(insertValueThatYouWishToPrintHere) to print on different lines.

In IntelliJ, a neat shortcut for the print statement is to type “sout” and press enter.

Please write the number of each task as a comment above each task. For example:

```
// 1
int myInt = 10;
// 2
int myInt2 = 4;
```

1. Declare and initialize a variable of type **int** called **myInt**. Initialize it to the value **10**.

```
// 1
int myInt = 10;
```

2. Declare and initialize a variable of type **int** called **myInt2**. Initialize it to the value **4**.
int myInt2 = 4;
3. Declare and initialize a variable of type **double** called **myDouble**. Initialize it to the value **2.5**.
4. Declare and initialize a variable of type **char** called **myChar**. Initialize it to the value **A**.
5. Print out the expression **myInt** divided by **myDouble**. What result do you get?

```
/Users/renita/Library/Java/JavaVirtualMachines/openjdk-19.0.1/Contents/Home/bin/java -
4.0

Process finished with exit code 0
```

6. Print out the expression **myInt** divided by **myInt2**. What result do you get?

2

Process finished with exit code 0

7. What did you learn from **tasks 5** and **6**?

We learned that an integer divided by a double will be a double and an integer by an integer would still be an integer and round down like computer science.

8. **Cast** the variable **myDouble** to an **int** and store it in a variable called **myInt3**.
9. Print the variable **myInt3**. What result do you get?

2

10. What *type* of casting is this an example of?

Manual.

11. Print the statement **12/0**. What result do you get?

```
Exception in thread "main" java.lang.ArithmeticException: Create breakpoint : / by zero
    at ArrayUtility.main(ArrayUtility.java:24)
```

12. Print the statement **12.0/0**. What result do you get?

Infinity

13. Declare a variable called **myDouble2** and initialize it to 4.6.
14. Declare a variable called **myDouble3** and initialize it to 4.4.

15. Declare a variable called **myDouble4** and initialize it to 4.5.

16. Add 0.5 to the variable called **myDouble2** using the compound addition operator.

17. Cast **myDouble2** to an **int**. Print this value. What result do you get?

5

18. Add 0.5 to the variable called **myDouble3** using the compound addition operator.

19. Cast **myDouble3** to an **int**. Print this value. What result do you get?

4

20. Add 0.5 to the variable called **myDouble4** using the compound addition operator.

21. Cast **myDouble4** to an **int**. Print this value. What result do you get?

5

22. What did you learn from **tasks 16 - 21**?

When casting the double to an int, it rounds down.

23. Cast **myChar** to an **int** and print this value. What result do you get?

65

24. Cast **myInt** to a **char** and print this value. What result do you get?

Nothing was printed

25. Declare and initialize a variable of type **int** called **myInt3** to the value of 7.

26. Print the variable **myInt3**. What result did you get?

7

27. Print **myInt3++**. What result did you get?

7

28. Print the variable **myInt3**. What result did you get?

8

29. Print **++myInt3**. What result did you get?

9

30. What did you learn from **tasks 27 - 29**?

When you write ++ before and after an int it adds 1 to its value. But the order matters as before with ++ front, the +1 increment wouldn't happen immediately, only the next print. And for the after with ++ at the back, the +1 increment would happen immediately.

31. Print **145%10**. What result did you get?

5

32. Print **178%10**. What result did you get?

8

33. What did you learn from **tasks 31 and 32**?

% 10, gets the last digit of a number

34. Print **10%2**. What result did you get?

0

35. Print **11%2**. What result did you get?

1

36. Print **12%2**. What result did you get?

0

37. What did you learn from **tasks 34 - 37**?

% #, finds the remainder of the first number divided by second number and you can determine an odd or even number from that.