**89. What are methods.**

1. Idea of methods:

a. Monolithic problems: Whole program is one.

b. Then program was divided into sub modules.

c. To reduce complexity of modules people using classes.

d. Method is a write term for Object Oriented Programming.

2. Syntax:

(signature of method)=> returnType methodName(parameter List)

{

//body of method

}

3. Example:

int max(int x, int y)

{

if(x > y)

{

return x;

} else

{

return y;

}

}

4. A main method is static it can called only static methods.

5. When method is called respective parameters are copied and operation in methods are only performed on copies of original variables (**only parameter passing method in java**).

**90. Writing Methods.**

1. Static methods can call only static methods.

**91. Passing objects as parameters.**

1. While passing object address of the object is passed.

2. Which modifies actual object.

3. A method can have return type as an object if it returns object.

**92. Practicing passing objects as parameters.**

**93. Parameter passing.**

1. The parameters of actual method is called **Formal parameters.**

2. The parameters of called method is called **Actual parameters.**

3. There is only one method of parameter passing that is content of actual parameter is copied into actual parameter.

4. Primitives are passed by values and objects are passed by reference.

5. Java has only one parameter passing.

**94. Student Challenge: Find Prime number.**

1. Local must be initialized before used.

2. Static and Instance variables are automatically initialized to their default value(0).

**95. Method Overloading.**

1. Writing two or more methods with same name but difference mut be in parameter list.

2. Either datatype of parameter must be different or number of parameters must be different.

3.**Polymorphism:** Same name but different behaviour.

4. You cannot have methods with same signature.

5. return type is not considered while comparing two methods.

**96. Practicing Method Overloading.**

**97. Student Challenge: Overloaded Validate-method.**

1. Overloaded method to calculate areas.

2. Overloaded method to reverse a int or array.

3. Overloaded method to validate name and age.

**98. Variable Arguments vargs(…)**

1. All parameters must be of same datatype.

2. Example:

void show(int…x)

{}

\*You can take mix of mandatory arguments and variable arguments.

3. We can as many arguments as we want in above code.

4. Variable arguments should be the last parameter.

5. You can’t take multiple variable arguments.

6. printf() of C language uses ellipse so it can take as many arguments we want.

7. We can use System.out.printf() in java.

8. vargs and array passing is similar but not same.

9. Can’t create vargs of array.

10. We can use variable arguments in main function parameters.

**99. Practicing variable arguments.**

**100. Student Challenge: Calculate Discount.**

1. Maximum of numbers using varargs

2. Sum of all elements using varargs

3. Calculate Discount using varargs

1. Integer.MIN\_VALUE;

Capital because it is final value.

**101. Command Line Arguments.**

1. Passing arguments while executing the file on command line.

2. For example: java a arg1 arg2 …;

3. java CommandTest hello;

Here jvm is using CommandTest as command line argument.

4. Command line arguments are accessed through main function.

**102. Student Challenge: Sum of Numbers from Command Line.**

1. Adding Numbers using Command Line.

2. Double.parseDouble() this method is used to convert string to double.

3. if (x.matches("[0-9\\.]"))

\\. because it will say only **.** can between digits.

**103. Recursion**

1. A method calling itself is called recursive method.

2. Recursion creates series of function waiting for last function return value so they can return.

3. It like rubber band.

4. The code after the recursive statement is execute after execution of after the recursive statement.

5. Recursion is not used in programming.

6. It takes extra time and extra space.

7. Recursion is used in problem solving. Math have recursion to solve the problem.