Selenium WebDriver Training

Methods and Object

Testleaf
Always Ahead

The Golden Circle

What

What are methods?

Members of the class that performs an action or verification (logic)
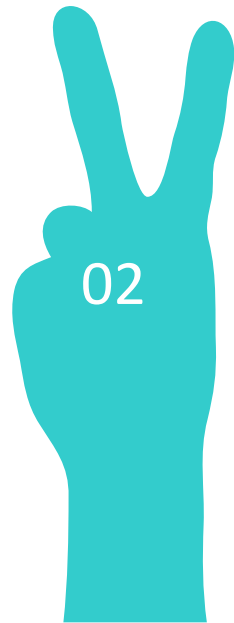
Why

Why are methods required?

Logically groups set of statements by functions

Testleaf
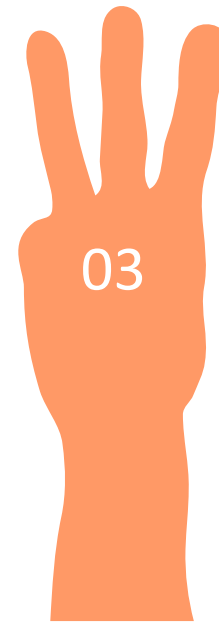Always Ahead

1.Access Modifier

Public   **Protected**   **Default (Package)**   **Private**

01   02   03   04

Testleaf
Always Ahead

1.Access Modifier

Access Modifier	Same Class	Same Package	Different Package	Different Project
Public	Yes	Yes	Yes	Yes
Protected	Yes	Yes	Yes	No
Default	Yes	Yes	No	No
Private	Yes	No	No	No

Testleaf
Always Ahead

1.Access Modifier (for Automation)

Generally, we use public as access modifier for most of the created methods so that it is accessible across projects

1.Access Modifier (for Automation)

Generally, we use public as access modifier for most of
the created methods so that it is accessible across projects

Testleaf
Always Ahead

Return type can be

- Primitive data types (int, boolean, char)
- Non Primitive data types (String)
- Void (keyword – that has nothing to return)

Testleaf
Always Ahead

2.Return Type

Return type	Keyword
Primitive Data types	return
Non-Primitive Data types	return
void	

Name of the method should be

- Meaningful
- (starts with)Lower case

It is the data needed for the method to work

- It can be no arguments, single or multiple arguments
- It can be primitive and non primitive and mix of both

Testleaf
Always Ahead

How to call a method

Step 1: Create a Object for the class (using new keyword)

Syntax :

ClassName object = new ClassName();

**Any meaningful name
(like short name)**

Testleaf
Always Ahead

Summary

- **Syntax :** access modifier – return type – method name – input parameters

- **Access Modifier:** public, protected, default, private

- **Return type :** should have return keyword for (non) primitive data

- **Input Parameter :** can be zero or more based on need

- **Object Creation :** using new keyword

- **Method call :** using Object created

Testleaf
Always Ahead

Classroom(Breakout)

● Create a new class Called Maths and
Write method syntax for add, multiply, divide of 2 numbers

● Before writing the code – follow the 3 step process:

- Understand the problem
- Solve the problem (Using Pseudocode)
- Write the syntax and then logic

● Call these methods from another class using object creation and confirm it is working

Testleaf
Always Ahead