**1.how jvm know the main method n explain, why main should be in static**

The main method is the entry point of the JVM when the class in launched. The JVM launches the Java program by invoking the main method of the class identified in the command to start the program. You can have more than one methods with name "main" but have different other signatures. But, the Java Virtual Machine can only be instructed to run one of them at a time.

When java runtime starts, there is no object of the class present. That's why the main method has to be static so that JVM can load the class into memory and call the main method. If the main method won't be static, JVM would not be able to call it because there is no object of the class is present.

**2.string reverse prgm**

**import** java.util.\*;

**public** **class** Reverse {

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

String value,reverse="";

Scanner in=**new** Scanner(System.***in***);

System.***out***.println("Enter the string: ");

value=in.nextLine();

StringBuilder input= **new** StringBuilder();

input.append(value);

input=input.reverse();

System.***out***.println("Reverse string: "+input);

}

}

(or)

**public** **class** Reverse {

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

String value,reverse="";

Scanner in=**new** Scanner(System.***in***);

System.***out***.println("Enter the string: ");

value=in.nextLine();

**int** length= value.length();

**for**(**int** i=length-1;i>=0;i--)

reverse=reverse+value.charAt(i);

System.***out***.println("Reverse String: "+reverse);

}

}

**3.about inheritance n uses**

Accessing one property class from another class using extends keyword.

Advantages code re-usability, memory wastage is low

Types of inheritance:

* Single
* Multiple
* Multilevel
* Hierarchical
* Hybrid

**4.how to find max salary in sql**

Select MAX(salary) from Employee

(or)

Select \* from DBO.Employee ORDERBY salary DESC

**8.d/b build( ) and perform( ) in actions class, can we use perform Without build?**

build()-method to perform multiple actions in single step, no need to use if we are performing single action.

perform()- method to perform actions without calling build

**9.methods in actions class**

Different Methods for performing Keyboard Events:

keyDown(modifier key): Performs a modifier key press.

sendKeys(keys to send ): Sends keys to the active web element.

keyUp(modifier key): Performs a modifier key release.

Different Methods for performing Mouse Events:

click(): Clicks at the current mouse location.

doubleClick(): Performs a double-click at the current mouse location.

contextClick() : Performs a context-click at middle of the given element.

clickAndHold(): Clicks (without releasing) in the middle of the given element.

dragAndDrop(source, target): Click-and-hold at the location of the source element, moves to the location of the target element

dragAndDropBy(source, xOffset, yOffset):  Click-and-hold at the location of the source element, moves by a given offset

moveByOffset(x-offset, y-offset): Moves the mouse from its current position (or 0,0) by the given offset

moveToElement(toElement): Moves the mouse to the middle of the element

release(): Releases the depressed left mouse button at the current mouse location