|  |  |
| --- | --- |
| **Compiler** | **Interpreter** |
| 1.Scan the entire program and translate it into machine code | 1.Translate program one statement at a time |
| 2.Compiler will take large amount of time to analyze the code | 2.Interpreter will takes less amount of time to analyze the code. |
| 3.However overall execution time and complexity is faster than interpreter | 3.However overall execution time and complexity is slower than compiler |
| 4.Generate object code which further queries lining hence required more memory | 4.No object code is generated hence are more efficient. |
| 5.Programing language like c, c++, java uses compiler. | 5.Programing like JavaScript, Ruby, and Python uses interpreter |
| 6.Compiler Follows Source Code to object code to Machine | 6.Interpreter follows Source Code to Intermediate Code to Interpreter |
| 7.It is best suited for the Production Environment | 7.It is best suited for the program and development environment. |
| 8.Compliers generates intermediate machine code | 8.Interpreter never generate any intermediate machine code |
| 9.Display all errors after, compilation, all at the same time | 9.Displays all errors of each line one by one |
|  |  |

**2)** **Few details of strongly typed language and loosely typed language minimum 4 examples?**

**A) Strong Type Language: -**

**1)** strongly typed language checks the type of a variable before performing an operation on it

**2)** A strongly typed language has stricter typing rules at compile time, which implies that errors and exceptions are more likely to happen during compilation

**3)** Most of these rules affect variable assignment, return values and function calling

**4)** For instance, Java is a strongly typed language because whenever you perform an operation on an object, Java checks the type of the object.

**Example:-** JAVA,Python can be both dynamically and strongly typed.

**Loosely Typed Language: -**

1. weakly typed language does not check the type of a variable before performing an operation on it
2. weakly typed languages perform implicit casts

Example: C Language, C++,.

1. **Work on data types? Write a Programs for Datatypes in java?**

**Source Code: -**

package prograday1;

public class DaataTypes {

public static void main(String[] args)

{

int num=10;

double numOne=20.0;

String str="poloju";

char ch='s';

float fot=1.1f;

boolean boll = true;

System.***out***.println(num);

System.***out***.println(numen);

System.***out***.println(strip);

System.***out***.println(ch);

System.***out***.println(fot);

System.***out***.println(boll);

}

}

**O/P:**

10

20.0

poloju

s

1.1

True

1. **Write the simplest code for the If, while ,for, switch?**

**For:**

package prograday1;

public class SampleForLoop {

public static void main(String[] args) {

String strOne="JAVA";

for(int i=0;i<=strOne.length()-1;i++)

{

for(int j=0;j<=i;j++)

{

System.***out***.print(strOne.charAt(j));

}

System.***out***.println(" ");

}

}}

**O/P:**

J

JA

JAV

JAVA

**While:**

package prograday1;

public class While {

public static void main(String[] args) {

int sumVal=2354, count=0;

while (sumVal!=0)

{

int n = sumVal % 10;

count = count + n;

sumVal = sumVal / 10;

}

System.***out***.println(count);

}

}

**O/P:**

14

**IF-Condition:**

package prograday1;

public class sample1 {

public static void main(String[] args) {

int year=2021;

if(year%400==0)

{

System.***out***.println(year+"leap year");

}

else if(year%100==0)

{

System.***out***.println(year+"not leap year");

}

else if(year%4==0)

{

System.***out***.println(year+"leap year");

}

else

{

System.***out***.println(year+"not a leap year");

}

}

}

O/P:

2021 not a leap year

**SWITCH Program:**

package prograday1;

public class Switch {

public static void main(String[] args) {

int number = 44;

String size;

switch (number) {

case 29:

size = "Small";

break;

case 42:

size = "Medium";

break;

case 44:

size = "Large";

break;

case 48:

size = "Extra Large";

break;

default:

size = "Unknown";

break;

}

System.***out***.println("Size: " + size);

}

}

**O/P:**

Size: Large

1. **In a class create constructor and object use data members and function members?**

**Source Code:**

package prograday1;

public class Constructor

{

public static int *ValOne*;

public static double *valTwo*;

public static String *str*;

public Constructor(int valOne,double valTwo,String str)

{

this.*ValOne*=valOne;

this.*valTwo*=valTwo;

this.*str*=str;

}

public void display()

{

System.***out***.println("integer "+*ValOne*+" double "+*valTwo*+" string "+*str*);

}

public static void main(String[] args)

{

Constructor c1 = new Constructor(10,10.0,"poloju");

c1.display();

}

}

O

**O/P:**

integer 10 double 10.0 string poloju

1. Create a class create 2 constructors in a single class use both parameterizes and non-parametrized?

**Source Code:**

package prograday1;

public class ConstructorTwo {

static String *str*;

public ConstructorTwo()

{

*str*="poloju";

}

public ConstructorTwo(String str) {

this.*str* = str;

}

public static void main(String[] args)

{

ConstructorTwo obj = new ConstructorTwo("poloju");

System.***out***.println(obj.*str*);

System.***out***.println(*str*);

}

}

**O/P:**

poloju

poloju