**NOTRE DAME UNIVERSITY BANGLADESH**



LAB Assignment 1

Object Oriented Programming

Name: Shazidul Alam

Subject: OOP

Student ID: 0692220005101009

Batch: CSE-19

**Lab-Topics Page**

**1) Writing simple java code like printing- “Hello CSE!”---------------------1**

**2) Variable Initializing and printing data: --------------------------------------1**

**a. Int,**

**b. Double,**

**c. Float,**

**d. Char,**

**e. String,**

**f. Short,**

**g. Long,**

**h. Boolean**

**3) Final keyword usage--------------------------------------------------------------3**

**4) Take input from keyboard and print the input (with or without newline)---------------------------------------------------------------------------------3**

**5) Print variable and text together ----------------------------------------------4**

**6) Array**

**a. Initialize array with given value-----------------------------------------------4**

**b. Initialize array with keyboard input------------------------------------------5**

**c. Take the array size input from keyboard------------------------------------5**

**d. One-dimensional array (input and print)-----------------------------------6**

**e. Two-dimensional array (input and print)-----------------------------------7**

**f. Three-dimensional array (input and print)---------------------------------8**

**g. Add two int array and keep the result in another array----------------9**

**7) Usage of every bitwise operator----------------------------------------10-11**

**8) String**

**a. Take input from keyboard and print the inputs-------------------------12**

**b. Initializing string inputs--------------------------------------------------------12**

**c. Find the string length-----------------------------------------------------------13**

**d. Convert string case- Upper to Lower and Lower to Upper-----------13**

**e. Find the index of a substring of a string-----------------------------------14**

**f. Replace a substring of string--------------------------------------------------14**

**g. Compare strings and print result--------------------------------------------15**

**h. Example:**

**i. Print the length of each string.------------------------------------------------16**

**ii. Replace all spaces of S1 to underscore(\_).--------------------------------16**

**iii. Print the first character of S1.-----------------------------------------------16**

**iv. Compare the string S1 and S2 and print “equal” or “not equal”**

**accordingly---------------------------------------------------------------------------17**

**v. Find the first occurrence of character ‘a’ in S1 and print it’s**

**position.-------------------------------------------------------------------------------18**

**vi. If S1 is a substring of S2 or S2 is a substring of S1 then print a**

**message.------------------------------------------------------------------------------18**

**vii. Convert the S1 string to lower case and S2 string to upper**

**case letter.----------------------------------------------------------------------------19**

**9) Mathematical Functions: abs(x), ceil, floor, round, power, square root, maximum and minimum of variables, pi etc.------------------------20**

**10) Usage of control statement: if-else, if-elseif, switch, ?, while, do-while, for, nested loop, continue, break--------------------------------------21**

**11) Method**

**a. A function with parameters and return type to find the maximum/minimum value------------------------------------------------------27**

**b. A function to add values taken from keyboard with no return type, no parameters ----------------------------------------------------------------------27**

**c. More than one functions which add, subtract, multiply and divides**

**given parameters and print the result without returning---------------28**

**d. Function with multiple parameters ----------------------------------------29**

**e. Function with void, double, int, string data type -----------------------29**

**f. Overloading method------------------------------------------------------------31**

**g. Ambiguous method-------------------------------------------------------------31**

**1) Writing simple java code like printing- “Hello CSE!”**

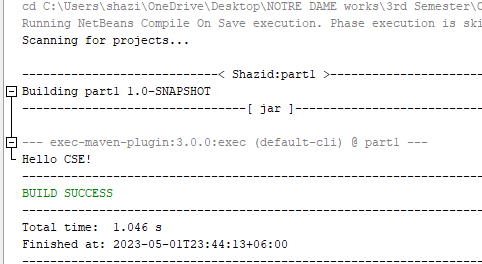
package shazid.part1;

public class Part1 {

public static void main(String[] args) {

System.out.println("Hello CSE!");

}}



**2) Variable Initializing and printing data:**

**a. Int, b. Double, c. Float, d. Char, e. String, f. Short, g. Long,**

**h. Boolean**

**Answer:**

package shazid.part2;

import java.util.Scanner;

public class Part2 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter an integer: ");

int Int = scanner.nextInt();

System.out.print("Enter a double: ");

double Double = scanner.nextDouble();

**1**

System.out.print("Enter a float: ");

float Float = scanner.nextFloat();

System.out.print("Enter a character: ");

char Char = scanner.next().charAt(0);

System.out.print("Enter a string: ");

String String = scanner.next();

System.out.print("Enter a short: ");

short Short = scanner.nextShort();

System.out.print("Enter a long: ");

long Long = scanner.nextLong();

System.out.print("Enter a boolean (true/false): ");

boolean Boolean = scanner.nextBoolean();

System.out.println("Int = " + Int);

System.out.println("Double = " + Double);

System.out.println("Float = " + Float);

System.out.println("Char = " + Char);

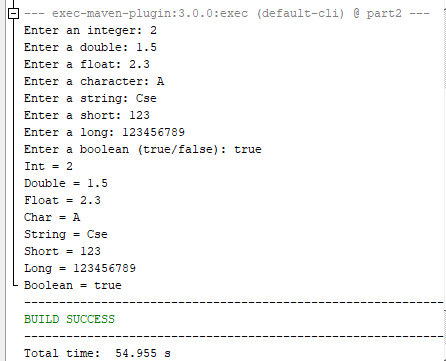
System.out.println("String = " + String);

System.out.println("Short = " + Short);

System.out.println("Long = " + Long);

System.out.println("Boolean = " + Boolean); }}

**2**



**3) Final keyword usage**

package shazid.part3;

public class Part3 {

public static void main(String[] args) {

final int MAX\_VALUE = 100;

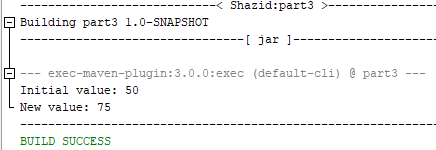
int value = 50;

System.out.println("Initial value: " + value);

value = 75;

System.out.println("New value: " + value);

}}



**4) Take input from keyboard and print the input (with or without newline)**

package shazid.part4;

import java.util.Scanner;

public class Part4 {

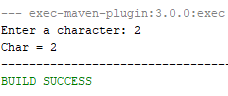
public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a character: ");

char Char = scanner.next().charAt(0);

System.out.println("Char = " + Char); }}



**3**

**5) Print variable and text together**

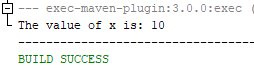
package shazid.part5;

public class Part5 {

public static void main(String[] args) {

int x = 10;

System.out.println("The value of x is: " + x); }}



**6. Array:**

**a. Initialize array with given value.**

public class Array\_intialization {

public static void main(String[] args) {

int a[]=new int[10];

a[0]=10;

a[1]=11;

a[2]=12;

a[3]=13;

a[4]=14;

a[5]=15;

System.out.println(a[0]);

System.out.println(a[1]);

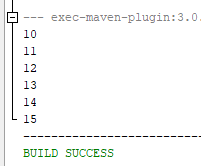
System.out.println(a[2]);

System.out.println(a[3]);

System.out.println(a[4]);

System.out.println(a[5]); }}

**4**



**b. Initialize array with keyboard input**

import java.util.\*;

public class Lab\_assi\_6b {

public static void main(String[] args) {

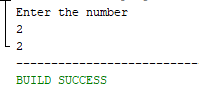
Scanner input= new Scanner(System.in);

int array[] = new int [100];

System.out.println("Enter the number");

(array[0])=input.nextInt();

System.out.println(array[0]); }}



**c. Take the array size from the input.**

import java.util.\*;

public class Lab\_assi\_6c {

public static void main(String[] args) {

int i;

Scanner input= new Scanner(System.in);

System.out.println("Enter the array size : ");

int size=input.nextInt();

System.out.println("Enter the number of inputs array will take : ");

**5**

int times=input.nextInt();

int array[]=new int [size];

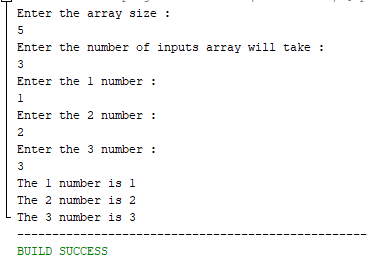
for( i=1; i<=times; i++)

{ System.out.println("Enter the "+i+" number : ");

array[i]=input.nextInt(); }

for(i=1; i<=times; i++)

{ System.out.println("The "+i+" number is "+array[i]); } } }



**d. One-dimensional array (input and print)**

*public class Array\_intialization*

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

*int a[]=new int[10];*

*a[0]=10;*

*a[1]=11;*

*a[2]=12;*

*a[3]=13;*

*a[4]=14;*

*a[5]=15;*

*System.out.println(a[0]);*

*System.out.println(a[1]);*

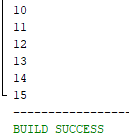
*System.out.println(a[2]);*

*System.out.println(a[3]);*

*System.out.println(a[4]);*

*System.out.println(a[5]);* } }

**6**



**e. Two-dimensional array (input and output).**

import java.util.\*;

public class Two-dimensional\_array\_(input\_and\_output) {

public static void main(String[] args) {

Scanner number=new Scanner(System.in);

int array[][]=new int[10][10];

System.out.println("\*\*\*\*\*Welcome dear students\*\*\*\*\*\n");

System.out.print("Enter the number of students\n");

int ID=number.nextInt();

System.out.print("Enter the number of subjects you have\n");

int subjects=number.nextInt();

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

{ for(int j=0;j<=subjects;j++) {

System.out.print("Enter the number of "+(i+1)+"students and "+(j+1)+"subject \n");

(array[i][j])=number.nextInt(); } }

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

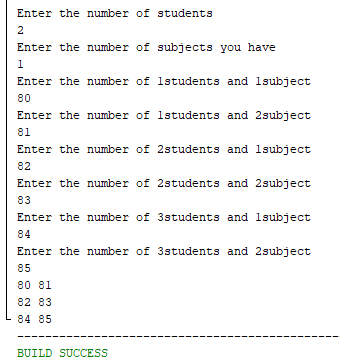
{ for(int j=0;j<=subjects;j++)

{ System.out.print(array[i][j]);

System.out.print(" "); }

System.out.print("\n"); }}}

**7**



**f. Three-dimensional array.**

import java.util.\*;

public class Lab\_assi\_6f {

public static void main(String[] args) {

int array[][][]=new int [2][5][7];

Scanner input=new Scanner(System.in);

System.out.println("Enter the number of lines : ");

int lines=input.nextInt();

for(int i=1; i<=lines; i++)

{ for(int j=1; j<=5; j++)

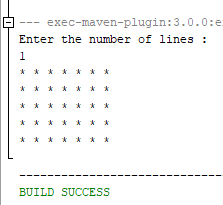
{ for(int k=1;k<=7; k++)

{ System.out.print("\* "); }

System.out.print("\n"); }

System.out.print("\n"); }}}

**8**



**g. Add two int array and keep the result in another array.**

public class Lab\_assi\_6g {

public static void main(String[] args) {

int i;

int array1[]=new int[20];

array1[0]=50;

array1[1]=70;

int array2[]=new int [20];

array2[0]=75;

array2[1]=69;

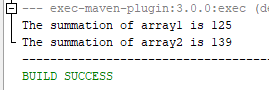
int arraysum[]=new int [20];

for( i=0; i<=1; i++)

{ arraysum[i]=array1[i]+array2[i]; }

for(i=0; i<=1; i++) {

System.out.println("The summation of array"+(i+1)+" is "+arraysum[i]); }}}



**9**

**7. Usage of these bitwise operators.**

**a) OR (|):** The OR operator performs a bitwise OR operation between two integer operands. Each bit of the result is 1 if either of the corresponding bits of the two operands is 1, otherwise, its 0.

For example:

int a = 3;

int b = 6;

int c = a | b;



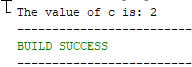
**b) AND (&):** The AND operator performs a bitwise AND operation between two integer operands. Each bit of the result is 1 only if both of the corresponding bits of the two operands are 1, otherwise, it's 0.

For example:

int a = 3;

int b = 6;

int c = a & b;

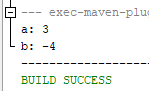


**c) NOT (~):** The NOT operator performs a bitwise NOT operation on a single integer operand. It inverts all the bits of the operand, so every 0 becomes 1 and every 1 becomes 0.

For example:

int a = 3;

int b = ~a;



**10**

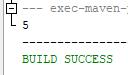
**d) XOR (^):** The XOR operator performs a bitwise XOR operation between two integer operands. Each bit of the result is 1 if the corresponding bits of the two operands are different, otherwise, it's 0.

For example:

int a = 3;

int b = 6;

int c = a ^ b;



**e) Shift Right (>>):** The shift right operator shifts all the bits of the left operand to the right by a specified number of bits. It inserts zeros into the most significant bits.

For example:

int a = 10;

int b = a >> 2;

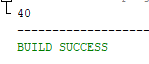


**f) Shift Left (<<):** The shift left operator shifts all the bits of the left operand to the left by a specified number of bits. It inserts zeros into the least significant bits.

For example:

int a = 10;

int b = a << 2;



**g) Shift Right Zero Fill (>>>):** The shift right zero fill operator works the same as the shift right operator, except that it always inserts zeros into the most significant bits, regardless of the sign of the operand.

For example:

int a = -10;

int b = a >>> 2;



**11**

**8. String**

**a. Take input from keyboard and print the inputs.**

Answer:

package shazid.part8a;

import java.util.Scanner;

public class Part8a {

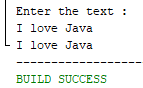
public static void main(String[] args) {

Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

String text=input.nextLine();

System.out.println(text); }}



**b. Initializing string inputs.**

package shazid.part8b;

import java.util.Scanner;

public class Part8b {

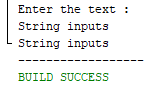
public static void main(String[] args) {

Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

String text=input.nextLine();

System.out.println(text); }}



**12**

**c. Find the string length**

Answer:

package shazid.part8c;

import java.util.Scanner;

public class Part8c {

public static void main(String[] args) {

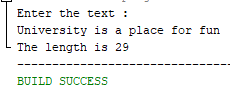
Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

String text=input.nextLine();

int length=text.length();

System.out.println("The length is "+length);



**d. Convert string case- Upper to Lower and Lower to Upper**

Answer:

package shazid.part8d;

import java.util.Scanner;

public class Part8d {

public static void main(String[] args) {

Scanner input= new Scanner(System.in);

System.out.println("Enter the text in completely upper case : ");

String text=input.nextLine();

String up=text.toLowerCase();

System.out.println("In lower case "+up);

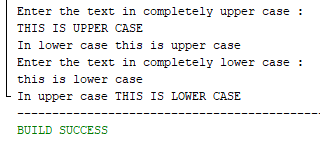
System.out.println("Enter the text in completely lower case : ");

String text1=input.nextLine();

String lower=text1.toUpperCase();

System.out.println("In upper case "+lower); }}

**13**



**e. Find the index of a substring of a string & f. Replace a substring of string**

Answer for e and f:

package shazid.part8e;

import java.util.Scanner;

public class Part8e {

public static void main(String[] args) {

Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

String text=input.nextLine();

String substring=text.substring(0, 5);

System.out.println("Main substring "+substring);

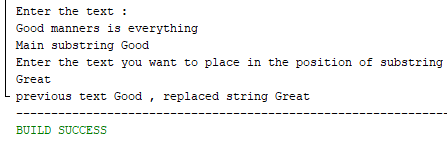
//now number f

System.out.println("Enter the text you want to place in the position of substring");

String replace=input.nextLine();

String main\_replace=substring.replaceAll(substring, replace);

System.out.println("previous text "+substring+", replaced string "+main\_replace); }}



**14**

**g. Compare strings and print result**

package shazid.part8g;

import java.util.Scanner;

public class Part8g {

public static void main(String[] args) {

Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

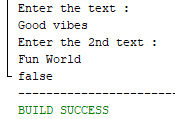
String text=input.nextLine();

System.out.println("Enter the 2nd text : ");

String text1=input.nextLine();

boolean judge=text.equals(text1);

System.out.println(judge); }}



**15**

**h. Example**:

**i. Print the length of each string. (i-vii)**

**ii. Replace all spaces of S1 to underscore(\_).**

import java.util.Scanner;

public class Lab\_assi\_8h {

public static void main(String[] args) {

Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

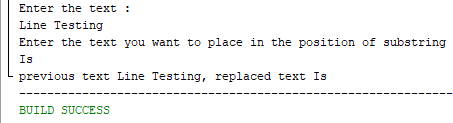
String text=input.nextLine();

System.out.println("Enter the text you want to place in the position of substring");

String replace=input.nextLine();

String main\_replace=text.replaceAll(text, replace);

System.out.println("previous text "+text+", replaced text "+main\_replace); }}



**iii. Print the first character of S1.**

import java.util.Scanner;

public class Lab\_assi\_8hiii {

public static void main(String[] args) {

Scanner input= new Scanner(System.in);

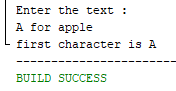
System.out.println("Enter the text : ");

String text=input.nextLine();

char first\_ch=text.charAt(0);

System.out.println("first character is "+first\_ch); }}

**16**



**iv. Compare the string S1 and S2 and print “equal” or “not equal” accordingly**

public class Comparing\_string {

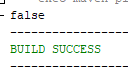
public static void main(String[] args) {

String s1="Shazid";

String s2="shazid";

boolean result=s1.equals(s2);

System.out.print(result); }}



Again,

package shazid.comparing\_string;

public class Comparing\_string {

public static void main(String[] args) {

String s1=" shazid ";

String s2=" shazid ";

boolean result=s1.equals(s2);

System.out.print(result); }}

C:\Users\shazi\AppData\Local\Microsoft\Windows\INetCache\Content.Word\part8iv2nd.png

17

**v. Find the first occurrence of character ‘a’ in S1 and print it’s position.**

import java.util.Scanner;

public class Lab\_assi\_8h\_v {

public static void main(String[] args) {

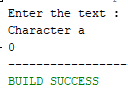
Scanner input= new Scanner(System.in);

System.out.println("Enter the text : ");

String text=input.nextLine();

int first\_ocr=text.indexOf(text, 0);

System.out.println(first\_ocr); }}



**vi. If S1 is a substring of S2 or S2 is a substring of S1 then print a message.**

public static void main(String[] args) throws Exception {

String s1 = "Java is fun";

String s2 = "is";

String s3 = "fun";

System.out.println("The first statement is "+s1);

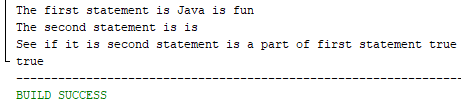
System.out.println("The second statement is "+s2);

String substr = (s1.substring(5, 7));

Boolean judge = (substr.equals(s2));

System.out.println("See if it is second statement is a part of first statement true if it matches false if it don't ");

System.out.println(judge); }



**18**

**vii. Convert the S1 string to lower case and S2 string to upper case letter.**

package shazid.part8vii;

import java.util.Scanner;

public class Part8vii {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter the text in completely upper case: ");

String text = input.nextLine();

String up = text.toLowerCase();

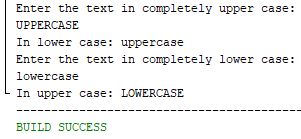
System.out.println("In lower case: " + up);

System.out.println("Enter the text in completely lower case: ");

String text1 = input.nextLine();

String lower = text1.toUpperCase();

System.out.println("In upper case: " + lower); }}



**19**

**9) Mathematical Functions: abs(x), ceil, floor, round, power, square root, maximum and minimum of variables, pi etc.**

package shazid.part9;

public class Part9 {

public static void main(String[] args) {

int x = -10;

double y = 3.1416;

double z = 2.5;

int a = 6;

int b = 12;

System.out.println("Absolute value of " + x + " is " + Math.abs(x));

System.out.println("Ceiling value of " + y + " is " + Math.ceil(y));

System.out.println("Floor value of " + y + " is " + Math.floor(y));

System.out.println("Rounded value of " + z + " is " + Math.round(z));

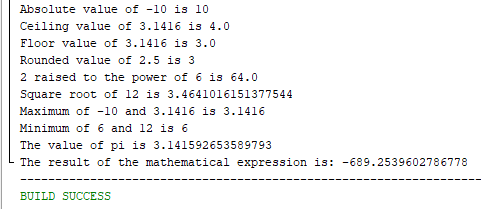
System.out.println("2 raised to the power of " + a + " is " + Math.pow(2, a));

System.out.println("Square root of " + b + " is " + Math.sqrt(b));

System.out.println("Maximum of " + x + " and " + y + " is " + Math.max(x, y));

System.out.println("Minimum of " + a + " and " + b + " is " + Math.min(a, b));

System.out.println("The value of pi is " + Math.PI);}}



**20**

**10)** **Usage of control statement: if-else, if-elseif, switch, ?, while, do-while, for, nested loop, continue, break.**

Answer:

**a) if-else:** The if-else statement is used to execute a block of code if a certain condition is true, and another block of code if the condition is false.

Example:

package shazid.part10a;

public class Part10a {

public static void main(String[] args) {

int x = 10;

int y = 5;

if (x > y) {

System.out.println("x is greater than y");

} else { System.out.println("y is greater than x"); }}}



**b) if-elseif:** The if-elseif statement is used to test multiple conditions and execute a different block of code for each condition.

Example:

package shazid.part10b;

public class Part10b {

public static void main(String[] args) {

int x = 5;

int y = 10;

if (x > y) {

System.out.println("x is greater than y");

} else if (x < y) {

System.out.println("y is greater than x");

} else {

System.out.println("x and y are equal"); }}}



**21**

**c) switch:** The switch statement is used to execute a different block of code for each possible value of a variable.

Example:

package shazid.part10c;

import java.util.Scanner;

public class Part10c {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("Enter a number between 1 and 7:");

int dayOfWeek = in.nextInt();

switch (dayOfWeek) {

case 1:

System.out.println("Saturday");

break;

case 2:

System.out.println("Sunday");

break;

case 3:

System.out.println("Monday");

break;

case 4:

System.out.println("Tuesday");

break;

case 5:

System.out.println("Wednesday");

break;

case 6:

System.out.println("Thursday");

break;

case 7:

System.out.println("Friday");

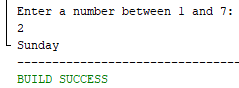
break;

default:

System.out.println("Invalid input. Please enter a number between 1 and 7.");

break; }}}

**22**



**d) ?:** The ternary operator ?: is a shorthand way of writing an if-else statement. It returns one value if the condition is true, and another value if the condition is false.

Example:

package shazid.part10d;

import java.util.Scanner;

public class Part10d {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.print("Enter the first number: ");

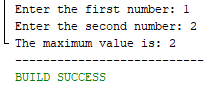
int x = in.nextInt();

System.out.print("Enter the second number: ");

int y = in.nextInt();

int max = (x > y) ? x : y;

System.out.println("The maximum value is: " + max); }}



**e) while:** The while loop is used to repeat a block of code while a certain condition is true.

Example:

package shazid.part10e;

public class Part10e {

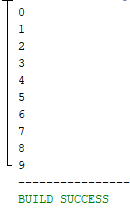
public static void main(String[] args) {

int x = 0;

while (x < 10) {

System.out.println(x);

x++; }}}



**23**

**f) do-while:** The do-while loop is similar to the while loop, but the block of code is executed at least once, even if the condition is false.

Example:

package shazid.part10f;

public class Part10f {

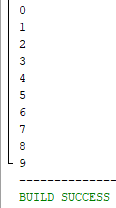
public static void main(String[] args) {

int x = 0;

do { System.out.println(x);

x++; }

while (x < 10); }}



**g) for:** The for loop is used to repeat a block of code for a certain number of times, or for each element in an array or collection.

Example:

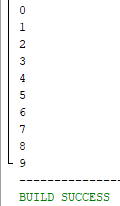
package shazid.part10g;

public class Part10g {

public static void main(String[] args) {

for (int i = 0; i < 10; i++) {

System.out.println(i); }}}



**24**

**h) nested loop:** A nested loop is a loop inside another loop. It's used to repeat a block of code for each possible combination of values in two or more variables.

Example:

package shazid.part10h;

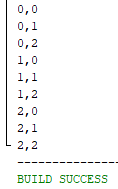
public class Part10h {

public static void main(String[] args) {

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

System.out.println(i + "," + j); }}}}



**i) continue:** The continue statement is used to skip the rest of the code in a loop and move on to the next iteration.

Example:

package shazid.part10i;

public class Part10i {

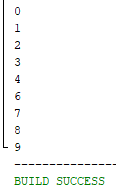
public static void main(String[] args) {

for (int i = 0; i < 10; i++) {

if (i == 5) {

continue; }

System.out.println(i); }}}



**25**

**j) break:** The break statement is used to exit a loop early, even if the condition is still true.

Example:

package shazid.part10j;

public class Part10j {

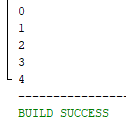
public static void main(String[] args) {

for (int i = 0; i < 10; i++) {

if (i == 5) {

break; }

System.out.println(i); }}}



**26**

**11) Method :**

**a. A function with parameters and return type to find the maximum/minimum value.**

Answer:

package shazid.part11a;

import java.util.Scanner;

public class Part11a {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter the First number : ");

int value1 = input.nextInt();

System.out.println("Enter the Second number : ");

int value2 = input.nextInt();

max(value1, value2); }

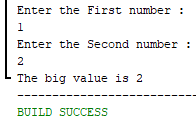
public static void max(int num1, int num2) {

if (num1 > num2)

System.out.println("The big value is " + num1);

else

System.out.println("The big value is " + num2); }}



**b. Function to add two numbers with no return type.**

package shazid.part11brenew;

import java.util.Scanner;

public class Part11bRenew {

public static void main(String[] args) {

getInputAndAdd(); }

public static void getInputAndAdd() {

try (Scanner input = new Scanner(System.in)) {

System.out.println("Enter the first integer: ");

int number1 = input.nextInt();

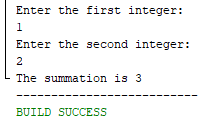
System.out.println("Enter the second integer: ");

int number2 = input.nextInt();

int sum = number1 + number2;

System.out.println("The summation is " + sum); }}}

**27**



**c. Function to make four function add, sub, multiply, division with no return type.**

package shazid.part11c;

import java.util.Scanner;

public class Part11c {

public static double add(double x, double y) {

return x + y; }

public static double multiply(double number1, double number2) {

return number1 \* number2; }

public static double sub(double value1, double value2) {

return value1 - value2; }

public static double division(double number1, double number2) {

return number1 / number2; }

public static void main(String[] args) {

try (Scanner input = new Scanner(System.in)) {

System.out.println("Enter the first number : ");

double num1 = input.nextDouble();

System.out.println("Enter the second number : ");

double num2 = input.nextDouble();

System.out.println("What do you want to do ?");

System.out.println("1. Addition");

System.out.println("2. Subtraction");

System.out.println("3. Multiplication");

System.out.println("4. Division");

int operation = input.nextInt();

if (operation == 1) {

System.out.println("The result is " + add(num1, num2));

} else if (operation == 2) {

System.out.println("The result is " + sub(num1, num2));

} else if (operation == 3) {

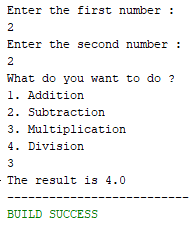
System.out.println("The result is " + multiply(num1, num2));

28

} else if (operation == 4) {

System.out.println("The result is " + division(num1, num2));

} else { System.out.println("Invalid input"); } } } }



**d. Function with multiple parameters.**

Answer:

package shazid.part11d;

public class Part11d {

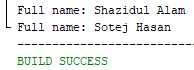
public static void main(String[] args) {

printFullName("Shazidul", "Alam");

printFullName("Sotej", "Hasan"); }

public static void printFullName(String firstName, String lastName) {

System.out.println("Full name: " + firstName + " " + lastName); }}



**e. Function with void, double, int, string data type**

Answer:

package shazid.part11e;

import java.util.Scanner;

public class Part11e {

public static void main(String[] args) throws Exception {

Scanner input=new Scanner(System.in);

System.out.println("What do you want to do \n1. personal info.\n2. sum.\n3. Average.\n4. String message.\n \tENTER KEY");

int value = input.nextInt();

switch (value) {

case 2: { System.out.println("Enter the first number ");

int num1 = input.nextInt();

System.out.println("Enter the second number ");

int num2 = input.nextInt(); **29**

int result =calculateSum(num1,num2);

System.out.println(result);

break; }

case 1: {

String name = "CSE 19";

int age = 22;

double height = 5.9;

printPersonInfo(name, age, height);

break; }

case 3: { System.out.println("Enter the first number ");

double num1 = input.nextDouble();

System.out.println("Enter the second number ");

double num2 = input.nextDouble();

double result = calculateAverage(num1,num2);

System.out.println(result);

break; }

case 4: {

printMessage("CSE 19"); }

default:

break; } }

public static void printPersonInfo(String name, int age, double height) {

System.out.println("Name: " + name);

System.out.println("Age: " + age);

System.out.println("Height: " + height + "feet"); }

public static int calculateSum(int num1, int num2) {

int sum = num1 + num2;

return sum; }

public static double calculateAverage(double num1, double num2) {

double sum = num1 + num2;

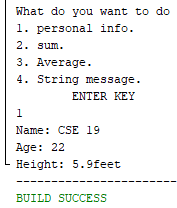
double average = sum/2;

return average; }

public static void printMessage(String message) {

System.out.println("Message: " + message); }}

**30**



**f) Overloading Method:**

Overloading methods enables you to define the methods with the same name as long as their signatures are different.

The max method that was used earlier works only with the int data type. But what if you need to determine which of two floating-point numbers has the maximum value? The solution is to create another method with the same name but different parameters, as shown in the following code:

package shazid.part11f;

public class Part11f {

public static void main(String[] args) {

int int1 = 5;

int int2 = 8;

double double1 = 10.2;

double double2 = 6.9;

System.out.println("Maximum between " + int1 + " and " + int2 + " is: " + max(int1, int2));

System.out.println("Maximum between " + double1 + " and " + double2 + " is: " + max(double1, double2));}

public static int max(int num1, int num2) {

if (num1 > num2) {

return num1;

} else {

return num2; }}

public static double max(double num1, double num2) {

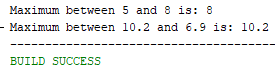
if (num1 > num2) {

return num1;

} else {

return num2; }}}

**31**



**g) Ambiguous Method:**

The Java compiler determines which method to use based on the method signature.

Sometimes there may be two or more possible matches for an invocation of a method, but the compiler cannot determine the most specific match. This is referred to as ambiguous invocation. Ambiguous invocation is a compile error.

Exampe:

public static void main(String[] args) {

System.out.println(max(1, 2)); }

public static double max(int num1, double num2){

if (num1>num2)

return num1;

else

return num2; }

public static double max(double num1, int num2){

if (num1>num2)

return num1;

else

return num2; }}



**32**

**THE END**