Object Oriented Programming – Exercise

Nama: Bryan Orville Audric

NIM: 2602160750

Kelas: LB20

1. No, to change the content of the string, it needs to be assigned to a string variable.
2. Source Code:

import java.util.Scanner;

class No2 {

static void check(char character){

if ((character >= '0' & character <= '9') || (character >= 'a' && character <= 'z') || (character >= 'A' && character <= 'Z')){

System.out.println(character +" is alphanumeric");

}else{

System.out.println(character +" is not alphanumeric");

}

}

public static void main(String args[] ) {

Scanner Input = new Scanner(System.in);

char Charactertest;

Charactertest = Input.next().charAt(0);

check(Charactertest);

Input.close();

}

}

1. Source Code:

import java.util.Scanner;

class No3 {

static void check(Character character){

if (Character.isUpperCase(character)){

System.out.println(character +" is UpperCase");

}else{

System.out.println(character +" is LowerCase");

}

}

public static void main(String args[] ) {

Scanner Input = new Scanner(System.in);

char Charactertest;

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

Charactertest = Input.next().charAt(0);

check(Charactertest);

Input.close();

}

}

1. Source Code:

import java.util.Scanner;

public class No4{

public static void main(String args[]){

Scanner Input = new Scanner(System.in);

System.out.println("Enter hexadecimal: ");

String hexadecimal = Input.next();

int DecimalNumber=Integer.parseInt(hexadecimal,16);

System.out.println(DecimalNumber);

}

}

Integer.parseInt() method parses the **string** argument which represents hexadecimal as a **decimal** integer object in the specified **parameter that specifies the number system to be used** by the second argument.

1. Source Code:

class No5{

public static void main(String args[]){

System.out.println(Math.pow(2,2));

System.out.println(Math.max(2,Math.min(3,4)));

System.out.println(Math.round(2.5F));

System.out.println(Math.ceil(-9.49));

System.out.println(Math.floor(7.5));

}

}

Output:

A number on a black background

Description automatically generated

1. Contains

Check whether a string contains a sequence of character.

Example:

String Stringtest = "OOP";

System.out.println(Stringtest.contains("O"));//Return True

System.out.println(Stringtest.contains("H"));//Return False

Concat

Concatenate specified string to the end of this string (string inside the argument)

Example:

//Concat

String FirstString = "Binus";

String SecondString = " University";

System.out.println(FirstString.concat(SecondString));//Output is “Binus University”

Compare to

Compare two strings lexicographically.

Example:

//Compare to

String String1 = "Binus";

String String2 = "Binus";

System.out.println(String1.compareTo(String2));//Return 0 since Both strings are equal

Format

Returns a formatted string using the specified locale, format string, and arguments.

Example:

//Format

String Word = "Food";

String Sentence = String.format("I eat %s", Word);

System.out.print(Sentence);//Output is “I eat food”

charAt

Return the character at the specified index.

Example:

//charAt

String Words = "Binus";

System.out.println(Words.charAt(0));//Output is “B”

Replace

Searches a string for a specified value and returns a new string where the specified values are replaced.

Example:

//Replace

String animal = "Duck";

System.out.println(animal.replace("D", "Y"));//Output is “Yuck”

Substring

Returns a new string which is the substring of a specified string.

Example:

//Substring

String FullString = "Binus University";

System.out.println(FullString.substring(0, 5));//Output is “Binus”

Trim

Removes whitespace from both ends of a string.

Example:

//Trim

String WordWithSpace = " Binus University ";

System.out.println(WordWithSpace.trim());// Output is “Binus University”

toCharArray

Convert string to a character array

Example:

//toCharArray

String MyString = "Binus";

char[] MyArray = MyString.toCharArray();

System.out.println(MyArray)// Output is “Binus”

Split

Split a string into an array of substrings.

Example:

//Split

String TwoWord = "Binus University";

String[] arrayofString = TwoWord.split(" ",2);

for(int i=0;i<2;i++){

System.out.println(arrayofString[i]);

}//Output is “Binus” and “University”

toLowerCase

Convert string to lowercase letter.

Example:

//toLowerCase

System.out.println("Binus University".toLowerCase());

//Output is “binus university”

toUpperCase

Convert string to Uppercase letter.

Example:

//toUpperCase

System.out.println("binus university".toUpperCase());

//Output is “BINUS UNIVERSITY”

1. In some cases, yes, but normally, you would convert this numeric values into the same type through casting
2. Source Code:

class No8{

public static void main(String args[]){

int a = 1;

double d = 1.0;

a = 46 % 9 + 4 \* 4 - 2;

a = 45 + 43 % 5 \* (23 \* 3 % 2);

a %= 3 / a + 3;

d += 1.5 \* 3 + (++a);

System.out.println(d);

}

}

Input/Result:



1. Yes, the following statements are correct.

Here is the output of each statement:

A black background with white text

Description automatically generated

1. Before the fixes:

class No10{

public static void main(String args[]){

int i; // variable i currently has no value.

int k = 100.0;

// variable k is declared as an int variable but not assigned to an int value

int j = i+1;

System.out.println("J is "+j + " and k is "+k);

}

}

After the fixes:

class No10{

public static void main(String args[]){

int i = 0; //Assigning variable i to a value

int k = 100;

//Assigning variable k to an int value

int j = i+1;

System.out.println("J is "+j + " and k is "+k);

}

}

1. These following conversions involving casting are allowed

Source code:

class No11{

public static void main(String args[]){

char c = 'A';

int i = (int)c;

System.out.println(i);

}

}

Output:



Source code:

class No11{

public static void main(String args[]){

float f = 1000.34F;

int i = (int)f;

System.out.println(i);

}

}

Output:



Source code:

class No11{

public static void main(String args[]){

double d = 1000.34;

int i = (int)d;

System.out.println(i);

}

}

Output:



Source Code:

class No11{

public static void main(String args[]){

int i = 97;

char c = (char)i;

System.out.println(c);

}

}

Output:



1. Source code:

class No12{

public static void main(String args[]){

int x = 1;

System.out.println((true)&& (3>4));//Print false

System.out.println(!(x>0)&& (x>0));//Print false

System.out.println((x!=1)==!(x==1));//Print true

System.out.println((x>=0)|| (x<0));//Print true

}

}

Output:

A black background with white text

Description automatically generated

1. Source Code:

class No13{

public static void main(String args[]){

System.out.println(2\*2-3>2&&4-2>5);

//Precedence of the Boolean above is:

// >

// >

// &&

System.out.println(2\*2-3>2||4-2>5);

//Precedence of the Boolean above is:

// >

// >

// ||

}

}

Output:

