Java module 1

Exercises Day 3

| 1.1 - While | Counting down |
| --- | --- |
| Instructions | Print the numbers counting down from 10 using a while loop |
| Expected output | 10  9  8  7  6  5  4  3  2  1  0 |
| Solution | public class Ex11 {  public static void main(String[] args) {    int counter = 10;  while (counter >= 0) {  System.out.println(counter);  counter--;  }  }  } |

| 1.2 - While | Enter the right password |
| --- | --- |
| Instructions | The user should enter a password, and it keeps asking until they enter the correct one. The right password is “password123” |
| Expected output | Enter the password:  >>>ABC123  Incorrect password. Try again.  Enter the password:  >>>password123  Password correct! |
| Solution | import java.util.Scanner;  public class Ex12 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);    System.out.print("Enter the password: ");  String password = scanner.nextLine();    while (!password.equals("password123")) {  System.out.println("Incorrect password. Try again");  System.out.print("Enter the password: ");  password = scanner.nextLine();  }  System.out.println("Password correct!");    scanner.close();  }  } |

| 1.3 - While | Guessing a number |
| --- | --- |
| Instructions | The program will select a random number between 1 and 100. The user will be asked to guess the number until he finds the right one. Each time, the program will tell him if the number is higher or lower.  hint: to choose a random number use Math.random()  <https://www.w3schools.com/java/java_howto_random_number.asp> |
| Expected output | Guess the number (1-100):  >>>33  Too low. Try again.  Guess the number (1-100):  >>>87  To high. Try again.  Guess the number (1-100):  >>>57  Congratulations! You guessed the number! |
| Solution | import java.util.Scanner;  public class Ex13 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  //Get a random number  int randomNum = (int)(Math.random() \* 101);  boolean continueGuessing = true;  while (continueGuessing) {  System.out.print("Guess the number (0 - 100): ");  int userGuess = scanner.nextInt();  if (userGuess < randomNum) {  System.out.println("Too small. Try again.");  } else if (userGuess > randomNum) {  System.out.println("Too big. Try again.");  } else {  System.out.println("You guessed!");  continueGuessing = false;  }  }    scanner.close();  }  } |

| 1.4 - While | Find the greatest number |
| --- | --- |
| Instructions | Write a program that reads numbers from the user until the user enters -1. The program should print the largest number entered. |
| Expected output | Enter any number:  >>> 33  Enter any number:  >>> -20  Enter any number:  >>> 57  Enter any number:  >>> -1  The greatest number was 57 |
| Solution | import java.util.Scanner;  public class Ex14 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter any number: ");  int number = scanner.nextInt();  int greatestNumber = number;  while (number != -1) {  System.out.print("Enter any number: ");  number = scanner.nextInt();  if (number > greatestNumber) {  greatestNumber = number;  }  }  System.out.print("The greatest number was: " + greatestNumber);  scanner.close();  }  } |

| 2.1 - String +While | Ask for a word starting with A |
| --- | --- |
| Instructions | Ask the user to type a word starting with the letter A. Check if it’s correct, otherwise ask again until it is correct. |
| Expected output | Enter a word starting with A:  >>> Kiwi  Incorrect!  Enter a word starting with A:  >>> Apple  Correct! |
| Solution | import java.util.Scanner;  public class Ex21 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  while (true) {  System.out.print("Enter a word starting with A: ");  String word = scanner.nextLine();  if (word.startsWith("A") || word.startsWith("a")) {  System.out.println("Correct!");  break;  } else {  System.out.println("Incorrect!");  }  }  scanner.close();  }  } |

| 2.2 - Strings | Uppercase or Lowercase? |
| --- | --- |
| Instructions | Ask the user a number, and a sentence. if the number is even, output the sentence in lowercase; otherwise, output it in uppercase.  Bonus: update your code such that you don’t use toLowerCase() and toUpperCase() functions. |
| Expected output | Enter a number:  >>> 7  Enter a sentence:  >>> Java is so Cool!  JAVA IS SO COOL! |
| Solution | import java.util.Scanner;  public class Ex22 {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter a number: ");  int number = scanner.nextInt();  scanner.nextLine();  System.out.print("Enter a sentence: ");  String sentence = scanner.nextLine();  if (number % 2 == 0) {  System.out.println(sentence.toLowerCase());  } else {  System.out.println(sentence.toUpperCase());  }  scanner.close();  }  } |
| Bonus solution | import java.util.Scanner;  public class Ex22 {  public static void main (String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter a number: ");  int number = scanner.nextInt();  scanner.nextLine();  System.out.print("Enter a sentence: ");  String sentence = scanner.nextLine();  int index = 0;  while (index < sentence.length()) {  char myChar = sentence.charAt(index); //Get the char  int intChar = (int)myChar; //Cast the char to a number  if (intChar > 64 && intChar < 91 && number % 2 == 0) {  intChar = intChar + 32; //Add 32 to get the number that corresponds to the lowercase char  }  if (intChar > 96 && intChar < 123 && number % 2 != 0) {  intChar = intChar - 32; //Subtract 32 to get the number that corresponds to the uppercase char  }  myChar = (char)intChar; //Cast the number back to a char  System.out.print(myChar);  index++;  }  scanner.close();  }  } |

| 3.1 - Casting | byte to short |
| --- | --- |
| Instructions | Create a byte variable with the value 100. Cast it to a short and print the value. Is it implicit or explicit casting? |
| Solution | byte myByte = 100;  short myShort = myByte;  System.out.println(myShort);  It is implicit casting. |

| 3.2 - Casting | long to int |
| --- | --- |
| Instructions | Create a long variable with the value 100000L. Cast it to an int and print the value. Is it implicit or explicit casting? |
| Solution | long myLong = 100000L;  int myInt = (int) myLong;  System.out.println(myInt);  It is explicit casting. |

| 3.3 - Casting | long to float |
| --- | --- |
| Instructions | Create a long variable with the value 100000L. Cast it to a float and print the value. Is it implicit or explicit casting? |
| Solution | long myLong2 = 100000L;  float myFloat = myLong2;  System.out.println(myFloat);  It is implicit casting. |

| 3.4 - Casting | What would the output be? |
| --- | --- |
| Instructions | short shortVal = 128;  byte byteVal = (byte) shortVal; // Explicit casting  System.out.println(byteVal);  Try it out. This conversion lead to data lost due to byte’s range of -128 to 127. |
| Solution | The output was -128. |