Huiswerk Native App Studio Week 1

Q2.

1. done

2.

A) private string birthplace;

B) public Person(String aName, int anAge, string aBirthplace)

{

name = aName;

age = anAge;

birthplace = aBirthplace;

}

C) Person ls = new Person("Luke Skywalker",34,"Alderaan");

D) System.out.println("and my birthplace is " + birthplace);

3. if (age>60) {

System.out.println("old person");

}

Toevoegen aan commentAboutAge

4. ls.commentAboutAge();

wp.commentAboutAge();

Toevoegen aan de main

5. public void growOlder() {

age += 1;

}

6. public void giveKnighthood() {

name = "Sir " + name;

}

7. System.out.println("Luke Skywalker’s age is " + ls.age);

Toevoegen aan de main

8. commentAboutAge();

Toevoegen aan de talk method

9. public void growOlderBy(int years) {

age = age + years;

}

oproepen in main via: ls.growOlderBy(10);

10. done

Q3.

1.done

2. public void salaryRise(int amount) {

salary = salary + amount;

}

oproepen in main via: e.salaryRise(10000);

3. public int netWorth() {

return savings - loan;

}

oproepen in main via: System.out.println("Ernies net worth = " + e.netWorth());

4. Omdat je in Java de output van Methodes kunt negeren als ze niet worden gebruikt of een verkeerde uitkomst hebben.

5. public String toString() {

return

"name = " + name + ",

savings = " + savings + ",

salary = " + salary + ",

loan = " + loan;

}

6. public Citizen(String aName) {

name = aName;

salary = 0;

savings = 0;

loan = 0;

}

Q4.

1. done

2. Flea pop = new Flea("Pop");

Flea squeak = new Flea("Squeak");

Flea zip = new Flea("Zip");

Toevoegen aan de main method.

3. Dog rex = new Dog("Rex", 8,pop);

Dog jimbo = new Dog("Jimbo",2,squeak);

Dog fido = new Dog("Fido", 6,zip);

Toevoegen aan de main method, na het aanmaken van de Fleas.

4. public String toString() {

return "my name is " + name + " and my age is " + age + " my flea is named " + dogsFlea.name;

}

voeg de volgende code toe aan het einde van de main method:

System.out.println(rex);

System.out.println(jimbo);

System.out.println(fido);

5. class DogOwner {

private String name;

private int salary;

private Dog ownersDog;

public DogOwner(String aName, int aSalary, Dog aDog) {

name = aName;

salary = aSalary;

dog = aDog;

}

}

6. voeg deze code toe aan het einde van de main method:

DogOwner angus= new DogOwner("Angus", 500,rex);

DogOwner brian = new DogOwner("Brian", 500,jimbo);

DogOwner charles = new DogOwner("Charles",500,fido);

7. public String toString() {

return "my name is " + name + " and my salary is " + salary + " my dogs name is " + ownersDog.name;

}

voeg deze code toe aan het einde van de main method:

System.out.println(angus);

System.out.println(brian);

System.out.println(charles);

8. dit vertelt je de naam van de vlo op de hond van die eigenaar, in dit geval: de vlo pop

Q5.

1. done

2.

public class SavingsAccount {

private int balance;

public SavingsAccount() {

balance = 0;

}

public SavingsAccount(int initialBalance) {

balance = initialBalance;

}

public void greet() {

System.out.println("Hello sir/madam");

}

public int showBalance() {

return balance;

}

public void deposit(int howMuch) {

if (howMuch < 0) {

System.out.println("You can’t deposit a negative amount!");

}

else {

balance = balance + howMuch;

}

}

public void withdraw(int howMuch) {

if (howMuch < 0) {

System.out.println("You can’t withdraw a negative amount!");

}

else {

balance = balance - howMuch;

}

}

public void transfer(SavingsAccount whereTo, int howMuch) {

balance = balance - howMuch;

whereTo.balance = whereTo.balance + howMuch;

}

}

Q6.

1. done

2. FredFlintstone.displayMe();

3. System.out.println("Barney's favourite color is " + BarneyRubble.favouriteColour);

4. CartoonCharacter fred = new CartoonCharacter("Fred Flintstone", "green", 30);

CartoonCharacter barney = new CartoonCharacter("Barney Rubble", "blue", 20);

CartoonCharacter wilma = new CartoonCharacter("Wilma Flintstone","pink", 30);

5. fred.displayMe();

barney.displayMe();

wilma.displayMe();

6. System.out.println("Barney ’s favourite color is " + barney.favouriteColour);

7. toevoegen aan de properties van de cartooncharacter: public static int count

Constructor veranderen naar dit:

public CartoonCharacter(String aName, String aColour, int aNumber) {

name = aName;

favouriteColour = aColour;

favouriteNumber = aNumber;

count +=1

}

dit toevoegen aan het einde van de main method: System.out.println(count);

8. System.out.println(Math.PI);

Q7.

1. done

2. er wordt altijd een waarschuwing gegeven, zelfs als er een positief getal uit komt.

3. Je moet een ; weghalen zodat het werkt.

4. Er wordt altijd een reset gedaan, ook al is de gegeven waarde positief.

5. Er zaten geen {} om de statements die binnen de IF functie hoorden.

6. public boolean isInRange(int value, int upperBound, int lowerBound) {

if (lowerBound <= value && value <= upperBound)

return true;

else

return false;

}

7. System.out.println(isInRange(30,100,30));

System.out.println(isInRange(50,100,30));

8. public boolean isInRangeIfLess(int value, int high, int low) {

return (low <= value && value <= high);

}

9. public boolean isInRange(int value, int high, int low) {

if (high <= low) {

System.out.println("The high value is lower than the low value ");

}

if (low <= value && value <= high)

return true;

else

return false;

}

Q8.

1. done

2. System.out.println(powerOf2A()); geeft 25

3. dit toevoegen binnen de functie: if (n == 0) return 1;

4. public void printLineA(int length) {

int i=0;

while (i < length) {

System.out.print("#"); i++;

}

System.out.println();

}

public void printLineB(int length) {

if (length == 0)

return;

int i=0;

do {

System.out.print("#"); i++;

}

while (i < length);

System.out.println();

}

Q9.

a. done

b. double[] nums = new double[10];

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

nums[i] =1+i\* 0.1;

}

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

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

}

f. public void printArray(double[] x) {

for (int i=0; i<x.lenght; i++) {  
 System.out.println(x[i]);

}

}

g. printArray(nums);

2. a done

b. Human[] threesome = new Human[3];

c. threesome[0] = new Human("Angus");

threesome[1] = new Human("Brian");

threesome[2] = new Human("Charles");

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

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

}

e. public void printArray(Human[] x) {

for (int i=0; i<x.length; i++) {

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

}

}

f. ArrayPrint.printArray(threesome);