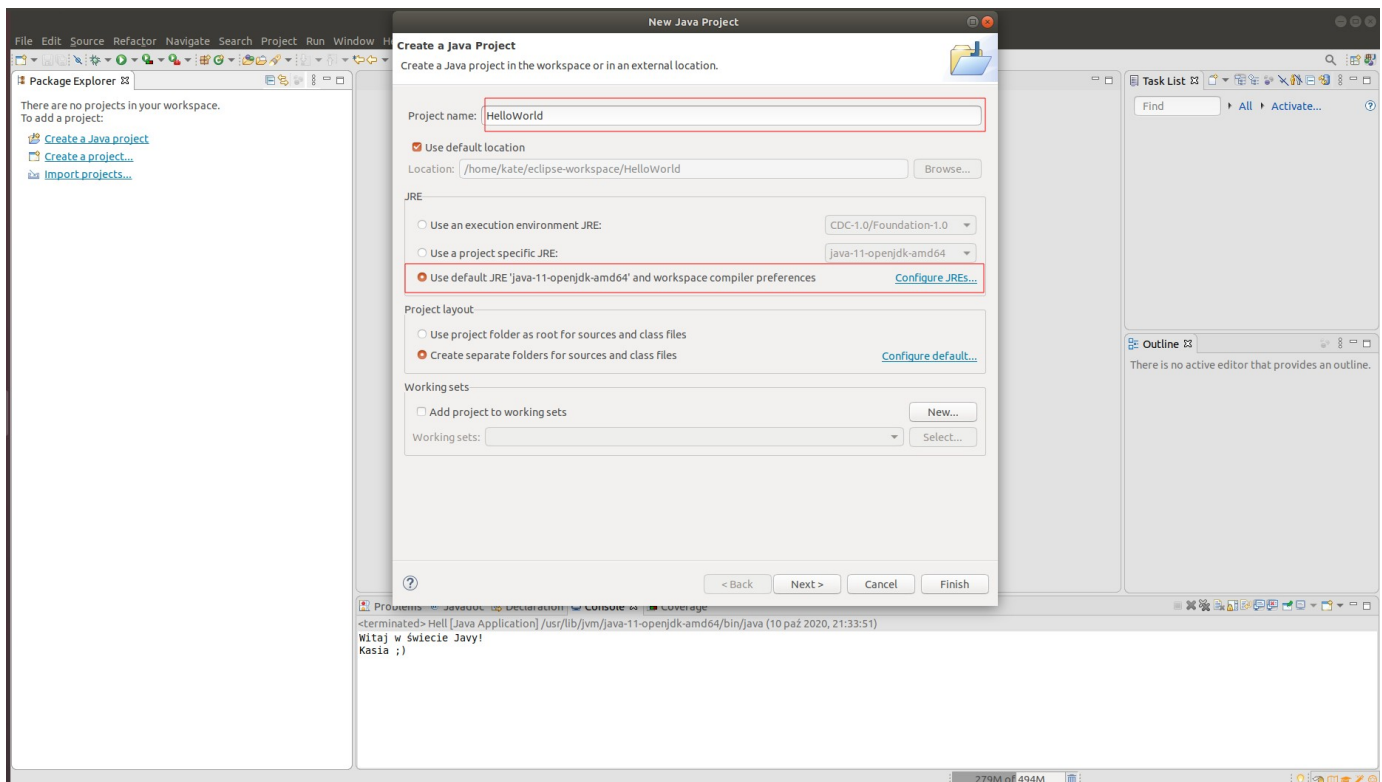
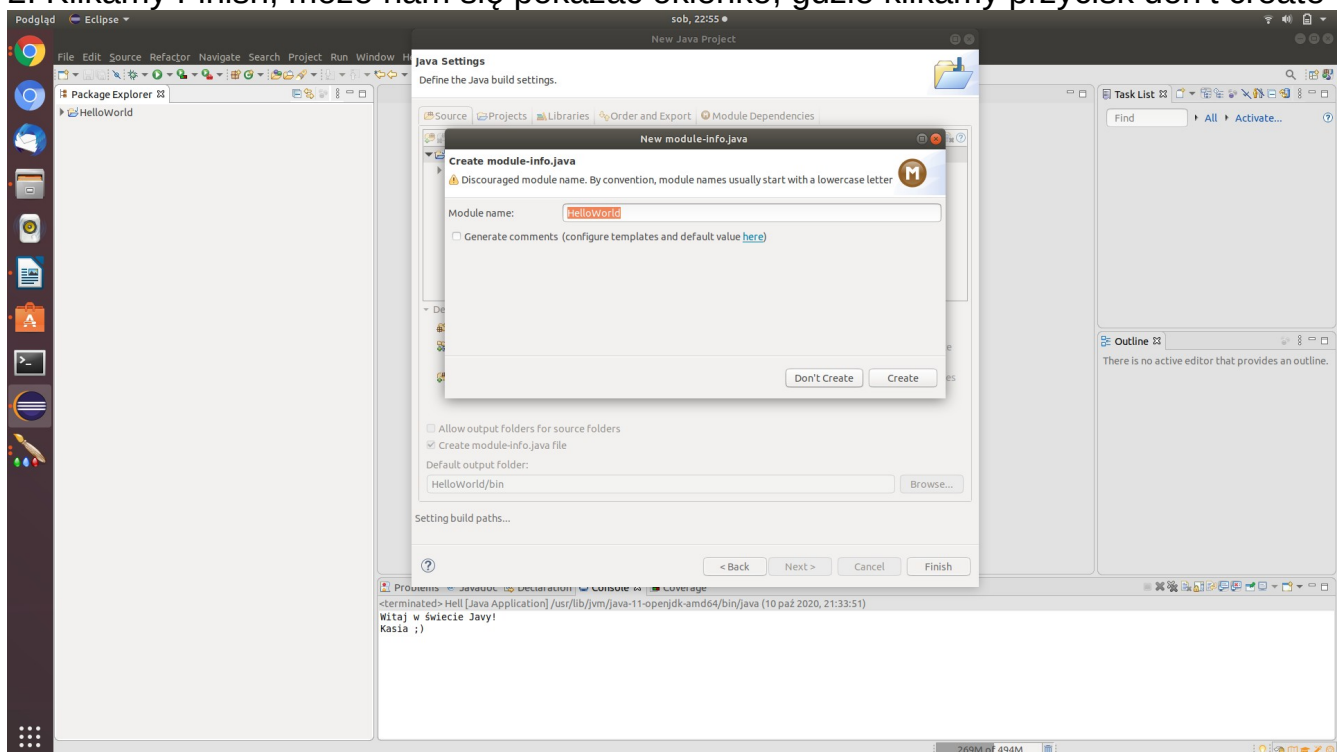


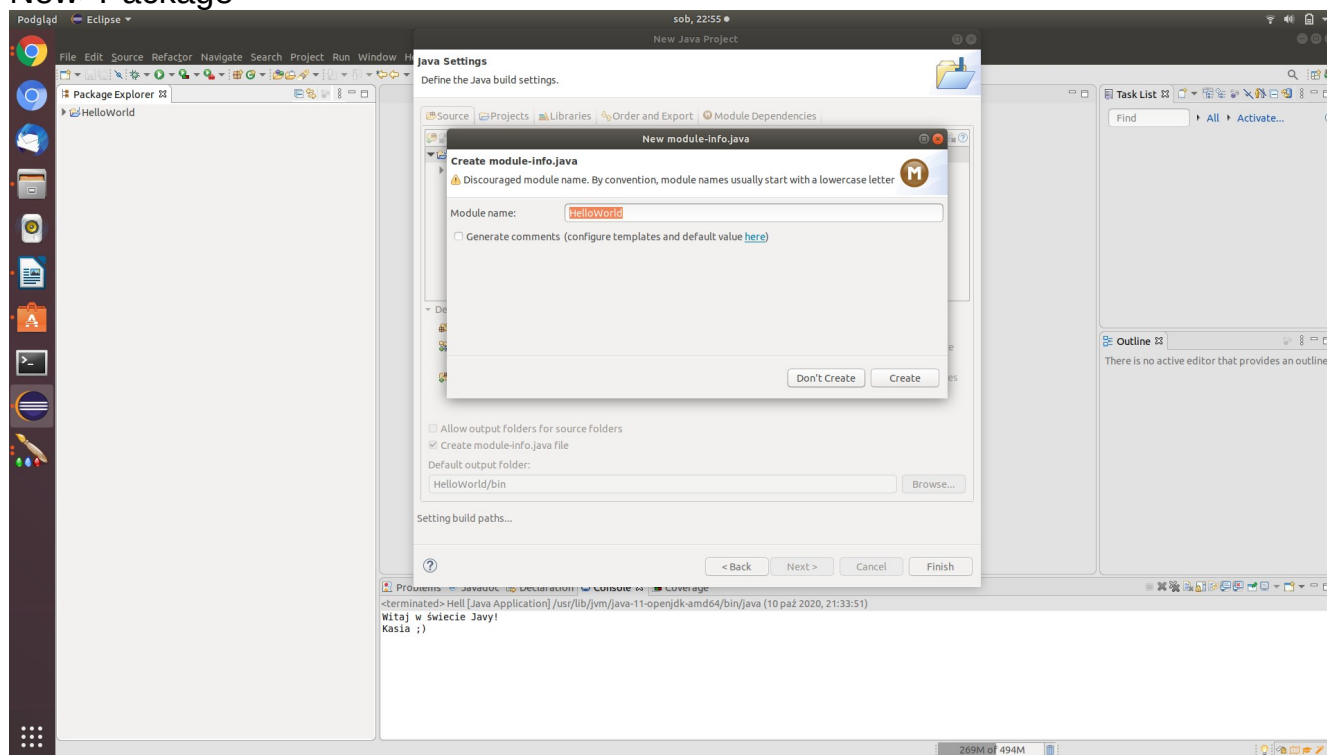
1. Tworzymy nowy projekt. File>New>Java Project Bardzo ważne! Pamiętaj aby zaznaczyć wyróżnione pole odnośnie JRE.



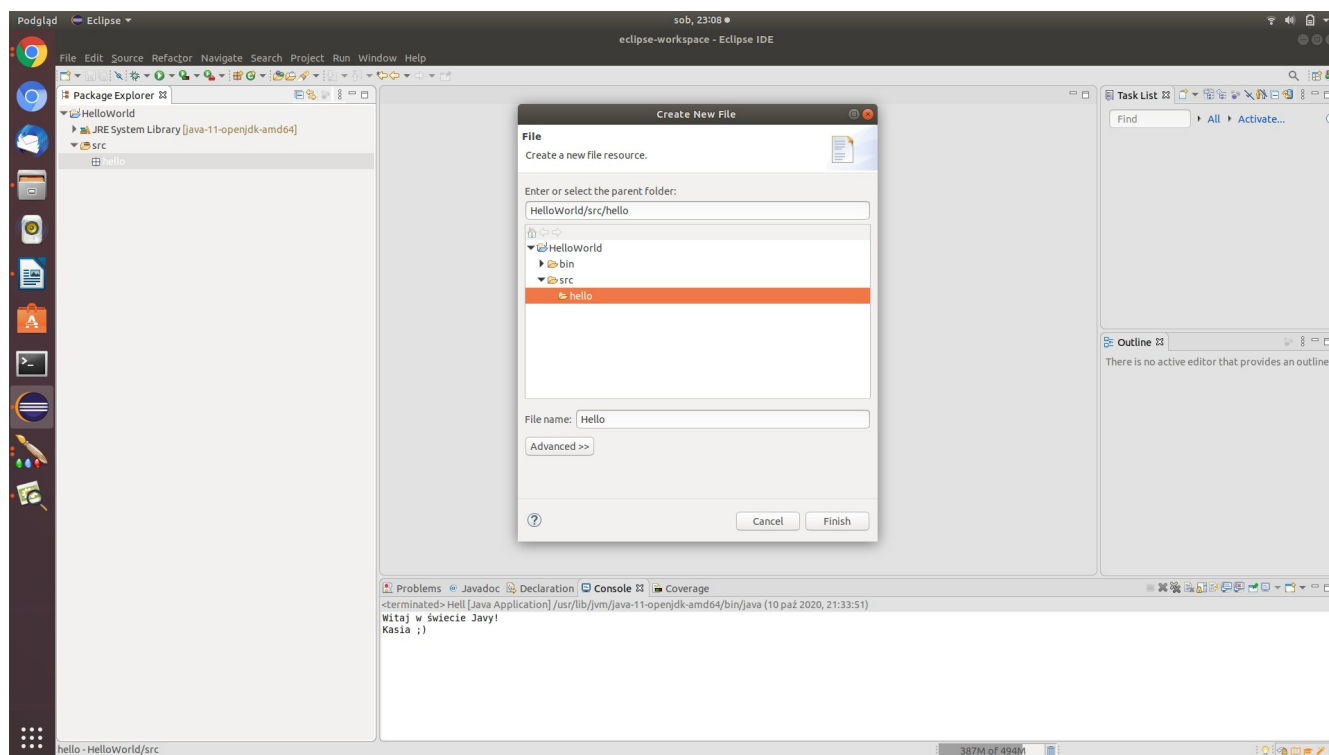
2. Klikamy Finish, może nam się pokazać okienko, gdzie klikamy przycisk don't create



3. W kolejnym kroku klikamy prawym przyciskiem myszy na nasz projekt i klikamy New>Package



4. W kolejnym kroku klikamy prawym przyciskiem myszy na nasz projekt i klikamy New>File, wpisujemy nazwę naszej klasy z rozszerzeniem .java (To bardzo istotne) i klikamy finish



5. W polu które się pojawiło wpisujemy nasz kod:

```
package hello;
import java.util.*;
import java.lang.*;
import java.io.*;

class Hello{
    public static void main(String []Args) {
        System.out.println("Witaj w świecie Javy!");
        System.out.println("Kasia ;");
    }
}
```

Wyjaśnienie:

Package hello tutaj wpisujemy naszą nazwę packagu, który utworzyliśmy w kroku 3. Import, tutaj dodajemy biblioteki. Część jest dostępna dzięki obecności jdk, natomiast w ten sam sposób można dołączyć nasz inny kod. W dużych projektach importujemy tutaj także klasy nadrzędne, a także inne klasy dzięki którym mamy dostęp do innych części programu.

Class to nasza klasa. Musi ona mieć dokładnie taką nazwę jak nasz plik java, ale bez rozszerzenia java. W naszym przypadku w pliku Hello.java mamy klasę Hello.

Public static void main(String[] Args) public to modyfikator dostępu. Takich modyfikatorów mamy trzy public, private i protected. Public oznacza że dostęp do metody main można mieć także spoza klasy Hello.

Static to metoda statyczna, oznacza to że ta metoda może się odnosić tylko i wyłącznie do innych metod które są statyczne.

Void to typ zwracany przez metodę. W tym przypadku typ pusty.

Nazwę metody zawsze piszemy małą literką. Dzięki temu nie udało (String[]Args) oznacza to że nasza metoda main zwraca tablicę stringów.

Ważne! W każdym programie musi istnieć metoda main i ona wygląda właśnie w ten sposób.

System.out.println(„”);- Jest to wydrukowanie na ekranie tekstu, wyniku lub co sobie tylko zamyślimy.

Pamiętajmy o średniku ; ;)

Kolejnym tematem były typy danych. W języku java występuje wiele typów danych, głównie można spotkać typy numeryczne takie jak(int, double, float), typy tekstowe(string, char), typ binarny(boolean), ale także typy złożone które omówimy później.

Przykład drugi.

Zamykamy nasz projekt klikając prawym przyciskiem myszy na projekt i klikając Close Project. Potem ponownie przechodzimy kroki 1,2,3 i 4 wpisując nowe wartości w nazwę projektu. Reszta może pozostać taka jak w poprzednim przykładzie.

```
package hello;

import java.util.*;
import java.lang.*;
import java.io.*;

class ZagubionyTelefon
{
    public static void main (String[] args)
    {
        int a = 5;
        int b = 6;
        int wynik;
        //System.out.println("Podaj a: ");
        //System.out.println("Podaj b: ");
        wynik = a * b;
        System.out.println("Wynikiem mnożenia jest:
"+wynik);
    }
}
```

W tym przykładzie wprowadzamy dwie zmienne o typie int. Typ int to typ liczbowy całkowity. Trzecią zmienną jest wynik który także jest o typie int.

System.out.println("Wynikiem mnożenia jest: "+wynik); w tej linijce wypisujemy tekst i dodajemy do niego naszą zmienną wynik.

Każda linijka zaczynająca się od // to komentarz stąd kompilator pomija te linijki.

Przykład trzeci.

Ponownie zamykamy otwarty projekt i tworzymy nowy projekt jak w przykładzie drugim.

```
package hello;

import java.util.*;
import java.lang.*;
import java.io.*;

class ZagubionyTelefon
{
    public static void main (String[] args)
    {
        int i;
        for(i=0; i<=10; i++)
        {
            System.out.println("Jestem zwycięzca");
        }
    }
}
```

Pętle.

W języku java wyróżniamy pętle takie jak:

if.. else, do... while, for.

W pętli for musimy zadeklarować nasz licznik, stąd mamy int i, które potem występuje w for(i=0;i<=10;i++). Pętla działa od i = 0, do i <= 10 z krokiem co 1 czyli i++.

W ciele pętli wypisujemy na ekranie 11 razy Jestem zwycięzcą.

Przykład czwarty.

Ponawiamy zamknięcie projektu z przykładu drugiego i założenie nowego projektu z przykładu pierwszego.

```
package hello;
import java.util.*;
import java.lang.*;
import java.io.*;

class ZagubionyTelefon
{
    public static void main (String[] args)
    {
        int i =0;
        int wynik;
        for(i=0; i<=10; i++)
        {
            wynik = i*2;
            System.out.println("Wynik =
"+wynik);
        }
    }
}
```

Tutaj ponownie wykorzystaliśmy pętlę for do pokazania że w pętli for możemy także wykonywać działania matematyczne.

Do domu

Prośba o poczytanie o poruszonych tematach i poćwiczcie jeszcze powyższe przykłady.

Na przyszłych zajęciach będziemy robić grę opartą o pętle.