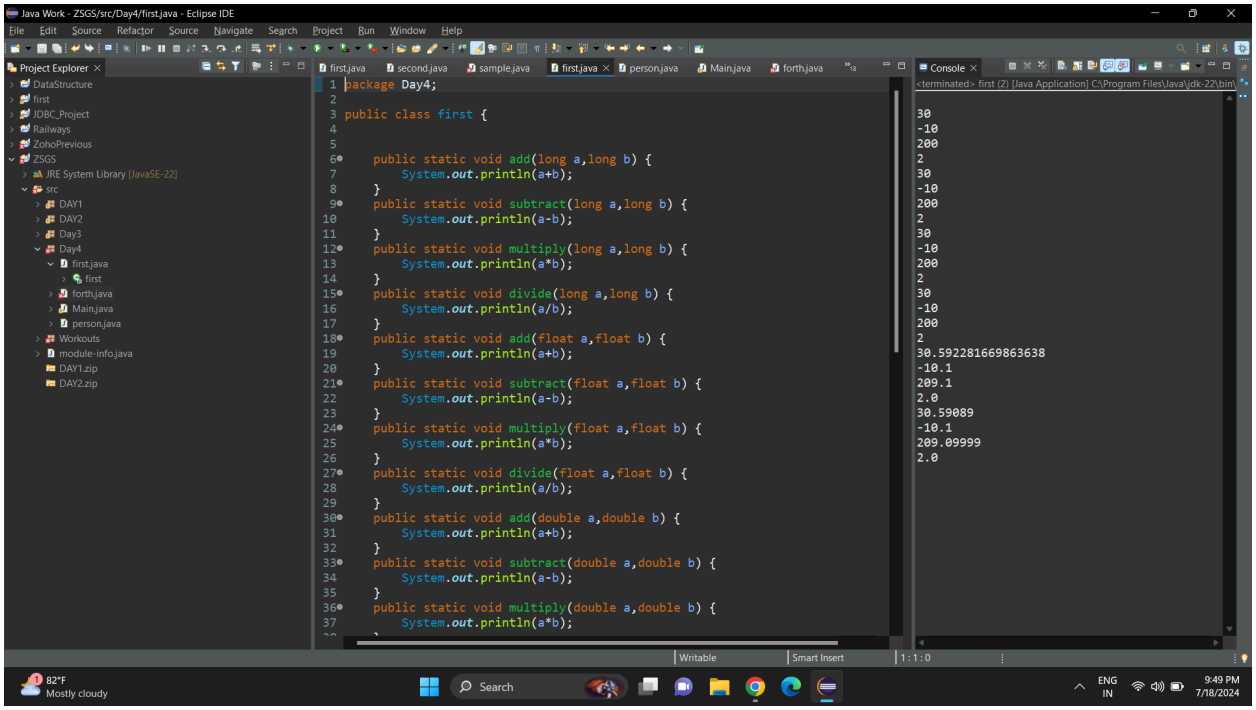
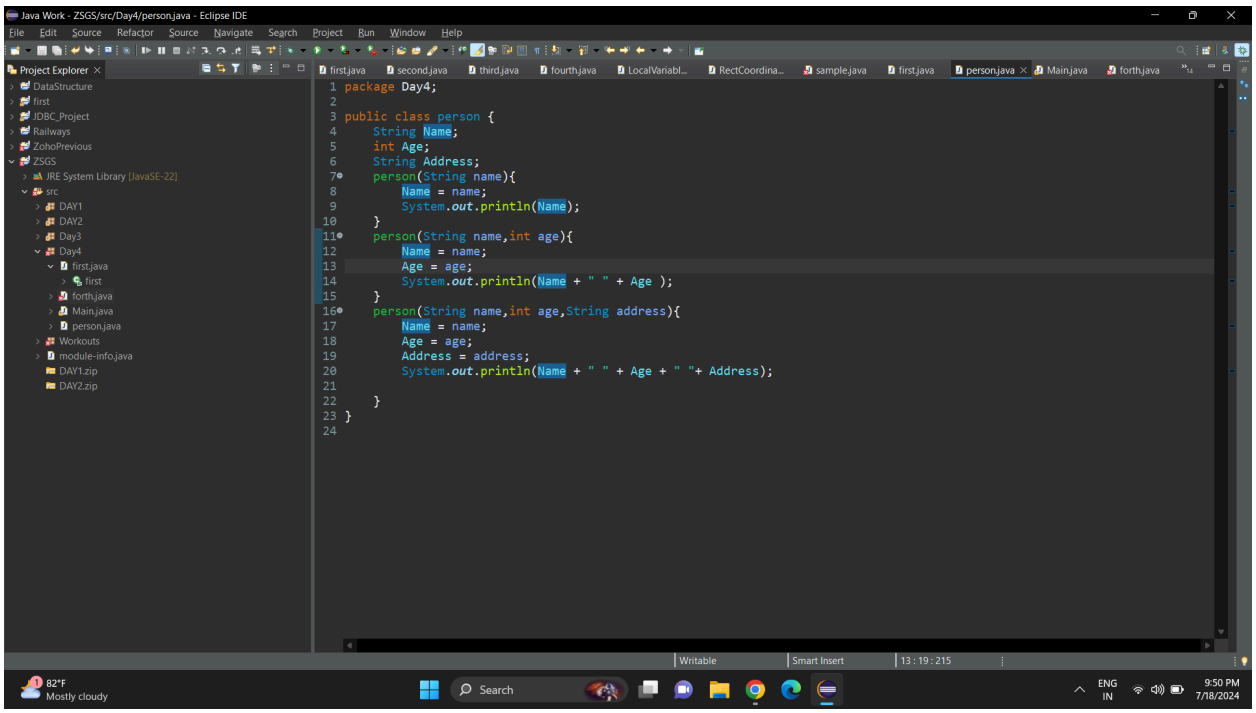
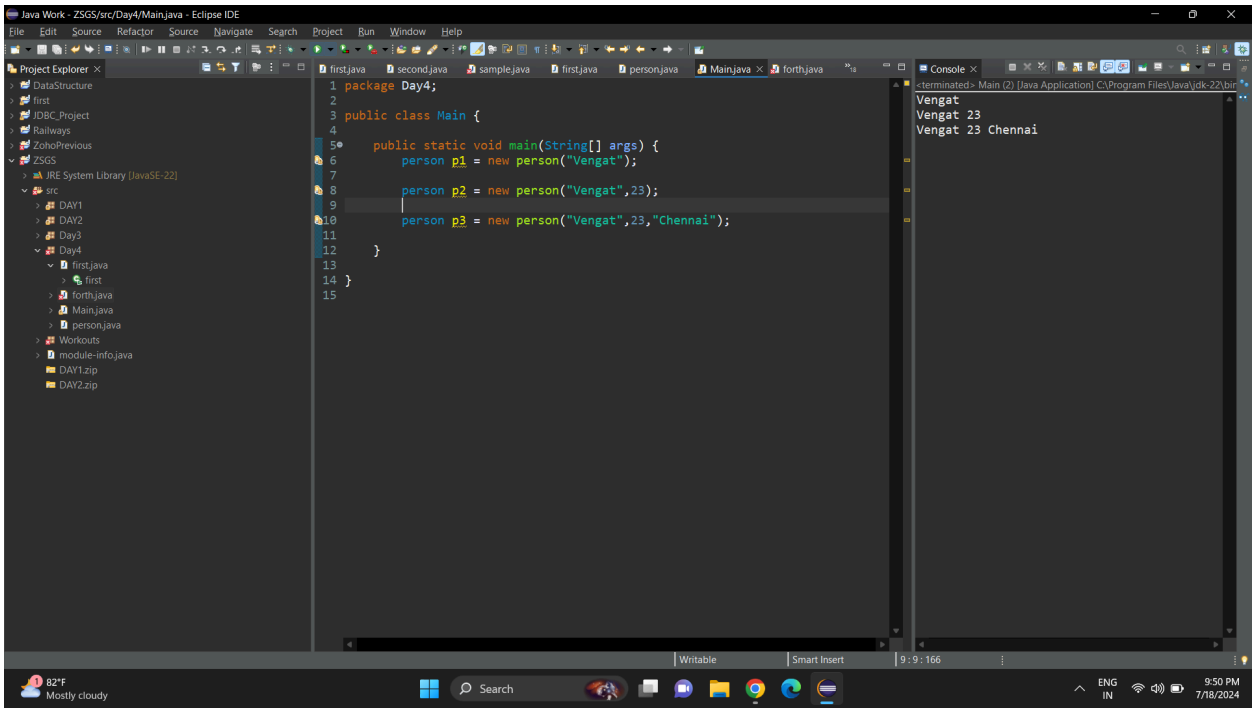
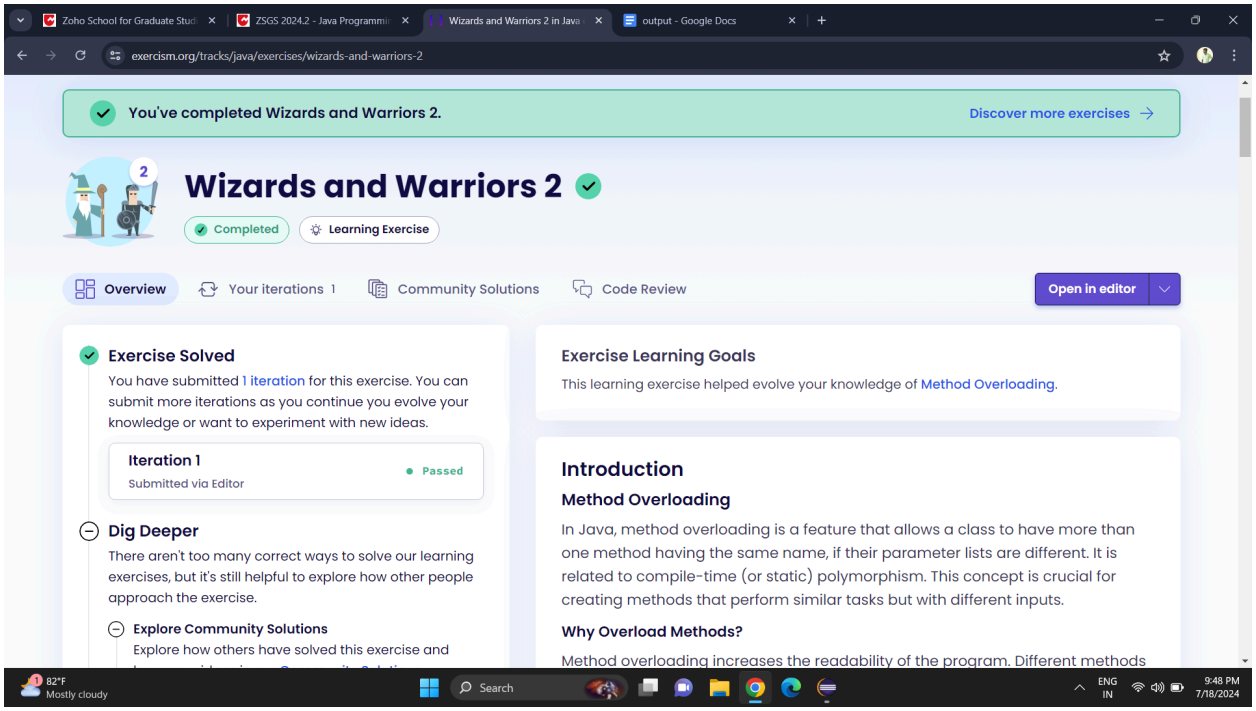

Java Work - ZSGS/src/Day4/first.java - Eclipse IDE
File Edit Source Refactor Source Navigate Search Project Run Window Help
Project Explorer
DataStructure
first
JDBC_Project
Railways
ZohoPrevious
ZSGS
JRE System Library [JavaSE-22]
src
DAY1
DAY2
Day3
Day4
first.java
first
forth.java
Main.java
person.java
Workouts
module-info.java
DAY1.zip
DAY2.zip
first.java
second.java
sample.java
first.java
person.java
Main.java
forth.java
package Day4;
public class first {
public static void add(long a,long b) {
System.out.println(a+b);
}
public static void subtract(long a,long b) {
System.out.println(a-b);
}
public static void multiply(long a,long b) {
System.out.println(a*b);
}
public static void divide(long a,long b) {
System.out.println(a/b);
}
public static void add(float a,float b) {
System.out.println(a+b);
}
public static void subtract(float a,float b) {
System.out.println(a-b);
}
public static void multiply(float a,float b) {
System.out.println(a*b);
}
public static void divide(float a,float b) {
System.out.println(a/b);
}
public static void add(double a,double b) {
System.out.println(a+b);
}
public static void subtract(double a,double b) {
System.out.println(a-b);
}
public static void multiply(double a,double b) {
System.out.println(a*b);
Console
<terminated> first (2) [Java Application] C:\Program Files\Java\jdk-22\bin\
30
-10
200
2
30
-10
200
2
30
-10
200
2
30.592281669863638
-10.1
209.1
2.0
30.59089
-10.1
209.09999
2.0
Writable
Smart Insert
1 : 1 : 0
82°F
Mostly cloudy
Search
ENG
IN
9:49 PM
7/18/2024


Java Work - ZSGS/src/Day4/person.java - Eclipse IDE
File Edit Source Refactor Source Navigate Search Project Run Window Help
Project Explorer
DataStructure
first
JDBC_Project
Railways
ZohoPrevious
ZSGS
JRE System Library [JavaSE-22]
src
DAY1
DAY2
Day3
Day4
first.java
first
forth.java
Main.java
person.java
Workouts
module-info.java
DAY1.zip
DAY2.zip
first.java
second.java
third.java
fourth.java
LocalVariabl...
RectCoordina...
sample.java
first.java
person.java
Main.java
forth.java
package Day4;
public class person {
String Name;
int Age;
String Address;
person(String name){
Name = name;
System.out.println(Name);
}
person(String name,int age){
Name = name;
Age = age;
System.out.println(Name + " " + Age );
}
person(String name,int age,String address){
Name = name;
Age = age;
Address = address;
System.out.println(Name + " " + Age + " "+ Address);
}
}
Writable
Smart Insert
13 : 19 : 215
82°F
Mostly cloudy
Search
ENG
IN
9:50 PM
7/18/2024


Java Work - ZSGS/src/Day4/Main.java - Eclipse IDE
File Edit Source Refactor Source Navigate Search Project Run Window Help
Project Explorer
DataStructure
first
JDBC_Project
Railways
ZohoPrevious
ZSGS
JRE System Library [JavaSE-22]
src
DAY1
DAY2
Day3
Day4
first.java
first
forth.java
Main.java
person.java
Workouts
module-info.java
DAY1.zip
DAY2.zip
first.java
second.java
sample.java
first.java
person.java
Main.java
forth.java
1 package Day4;
2
3 public class Main {
4
5 public static void main(String[] args) {
6 person p1 = new person("Vengat");
7
8 person p2 = new person("Vengat",23);
9
10 person p3 = new person("Vengat",23,"Chennai");
11
12 }
13
14 }
15
Console
<terminated> Main (2) [Java Application] C:\Program Files\Java\jdk-22\bir
Vengat
Vengat 23
Vengat 23 Chennai
Writable
Smart Insert
9 : 9 : 166
82°F
Mostly cloudy
Search
ENG
IN
9:50 PM
7/18/2024


Zoho School for Graduate Stud
ZSGS 2024.2 - Java Programmir
Wizards and Warriors 2 in Java
output - Google Docs
exercism.org/tracks/java/exercises/wizards-and-warriors-2
You've completed Wizards and Warriors 2.
Discover more exercises
Wizards and Warriors 2
Completed
Learning Exercise
Overview
Your iterations 1
Community Solutions
Code Review
Open in editor
Exercise Solved
You have submitted 1 iteration for this exercise. You can submit more iterations as you continue you evolve your knowledge or want to experiment with new ideas.
Iteration 1
Submitted via Editor
Passed
Dig Deeper
There aren't too many correct ways to solve our learning exercises, but it's still helpful to explore how other people approach the exercise.
Explore Community Solutions
Explore how others have solved this exercise and
Exercise Learning Goals
This learning exercise helped evolve your knowledge of Method Overloading.
Introduction
Method Overloading
In Java, method overloading is a feature that allows a class to have more than one method having the same name, if their parameter lists are different. It is related to compile-time (or static) polymorphism. This concept is crucial for creating methods that perform similar tasks but with different inputs.
Why Overload Methods?
Method overloading increases the readability of the program. Different methods
82°F
Mostly cloudy
Search
ENG
IN
9:48 PM
7/18/2024