Software Engineering
Design – 2024

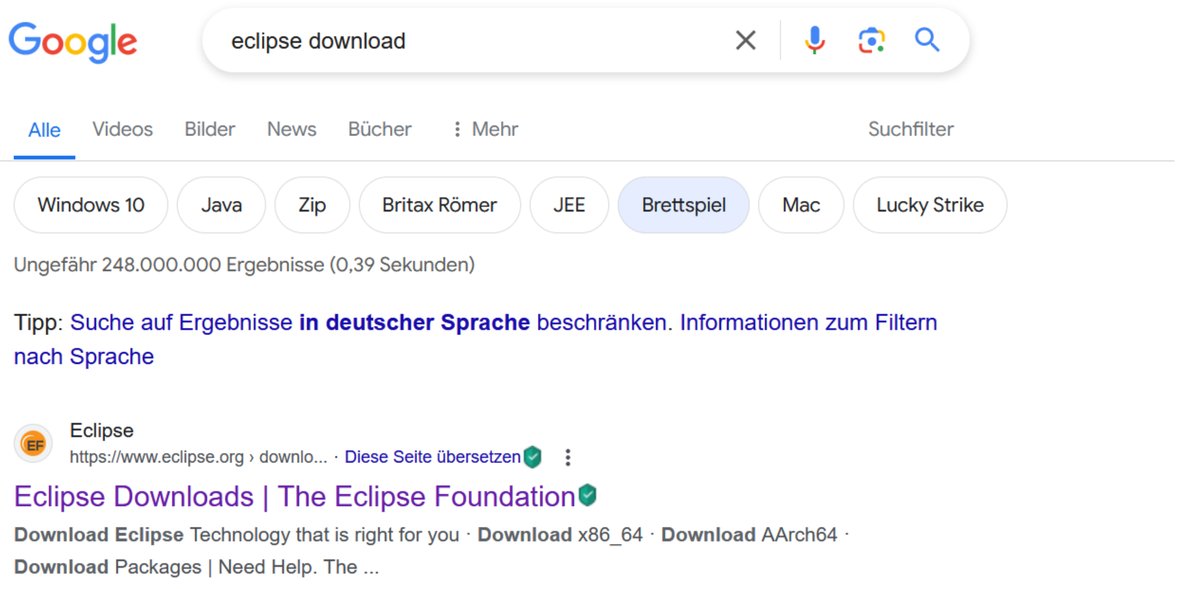
Lekoubou Tiwo Francis

1530478

Exercise 1 Documentation

1. IDE installation with screenshot

I am going to install eclipse for java developers. First type in google " download eclipse" and you will see the following overview like in Screenshot 1.



Ungefähr 248.000.000 Ergebnisse (0,39 Sekunden)

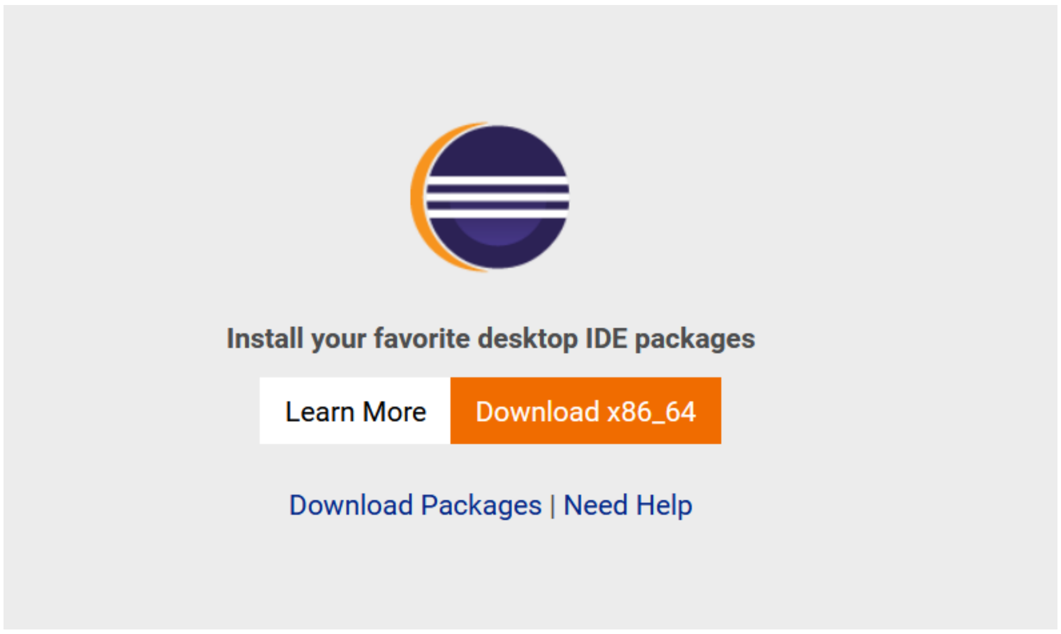
Tipp: Suche auf Ergebnisse **in deutscher Sprache** beschränken. Informationen zum Filtern nach Sprache

Eclipse
https://www.eclipse.org › downlo... · Diese Seite übersetzen

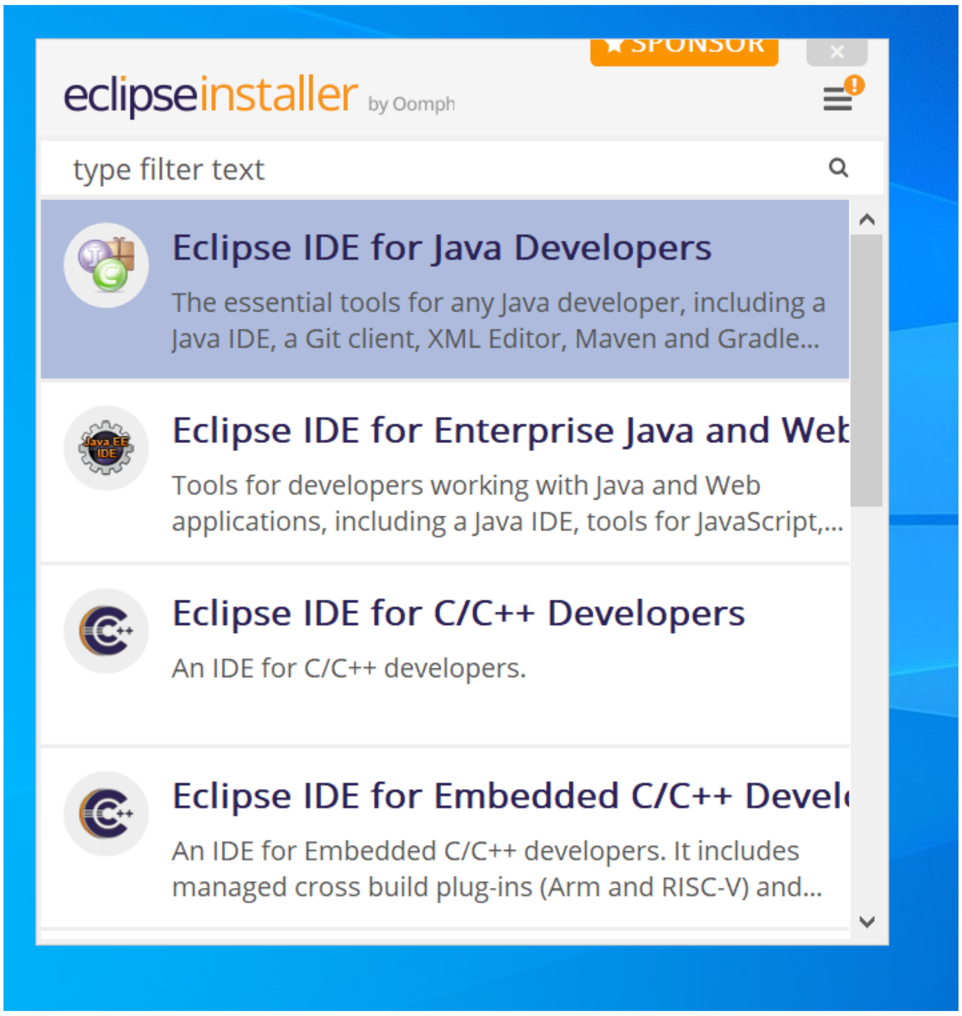
Eclipse Downloads | The Eclipse Foundation

Download Eclipse Technology that is right for you · **Download** x86_64 · **Download** AArch64 · **Download** Packages | Need Help. The ...

When this is open, you need to open the first link and go to the official site of Eclipse. When you are there, just click download. It should look like in the following screenshot.


Install your favorite desktop IDE packages
Learn More
Download x86_64
Download Packages | Need Help

After clicking download, you will see the following overview. Just choose “Eclipse IDE for Java Developers”.



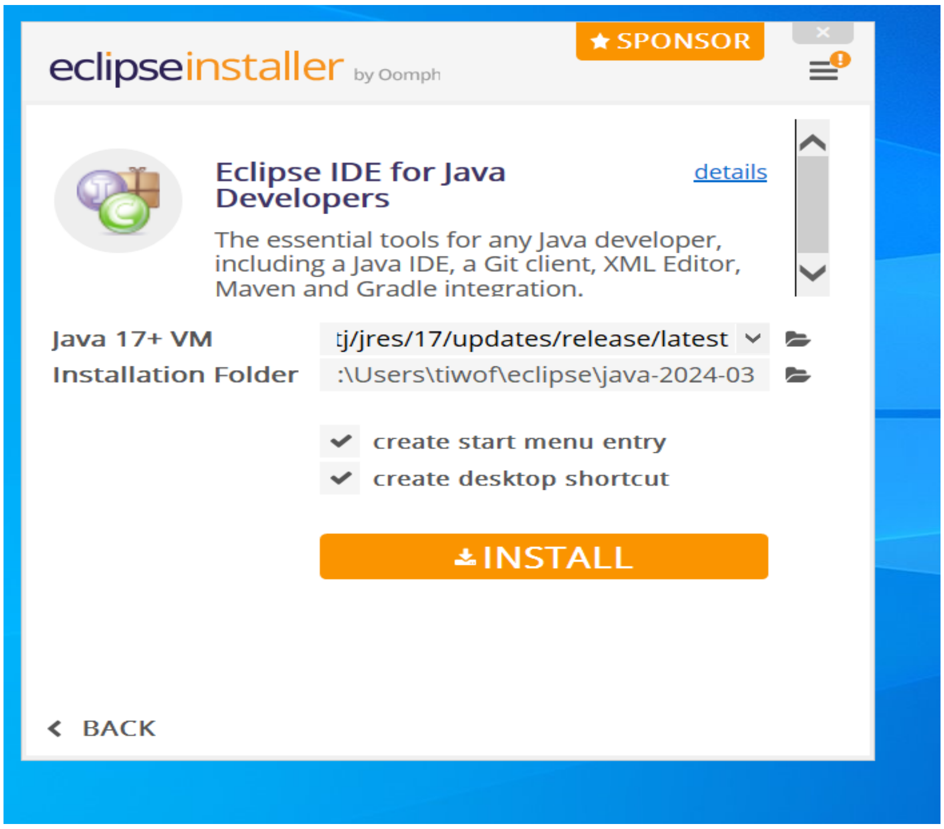
Eclipse IDE for Java Developers
The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle...

Eclipse IDE for Enterprise Java and Web
Tools for developers working with Java and Web applications, including a Java IDE, tools for JavaScript,...

Eclipse IDE for C/C++ Developers
An IDE for C/C++ developers.

Eclipse IDE for Embedded C/C++ Devel
An IDE for Embedded C/C++ developers. It includes managed cross build plug-ins (Arm and RISC-V) and...

As next you have the possibility to change the folder where you want to store the installation. I just leave everything as default and select install .



Eclipse IDE for Java Developers

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle integration.

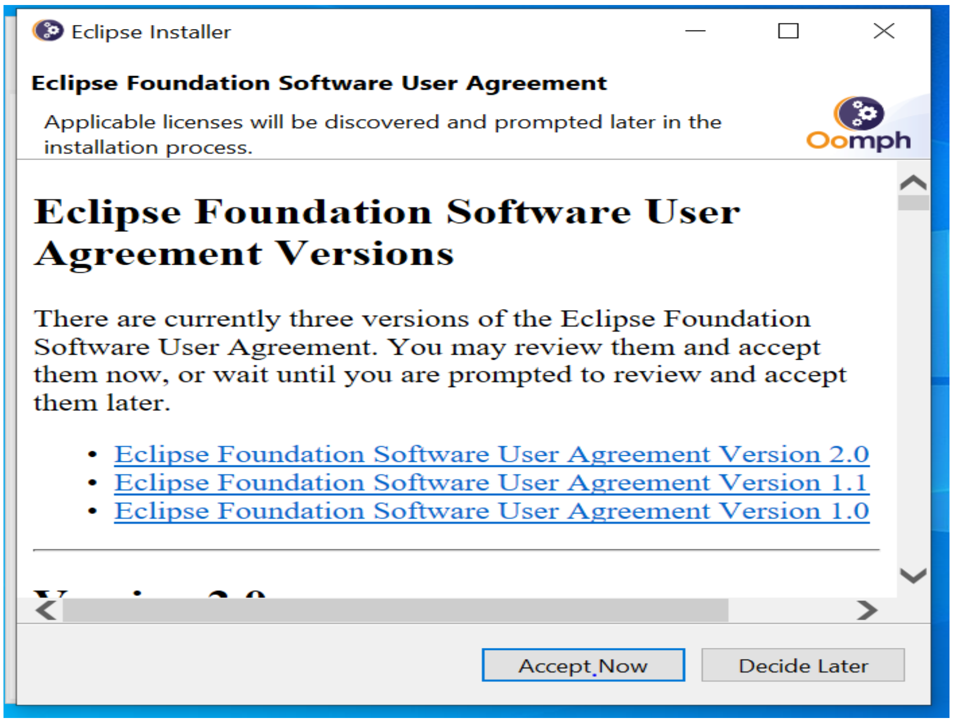
Java 17+ VM: tj/jres/17/updates/release/latest

Installation Folder: :\Users\tiwof\eclipse\java-2024-03

✓ create start menu entry

✓ create desktop shortcut

Now you just need to “read” and accept the terms and conditions.



Eclipse Foundation Software User Agreement

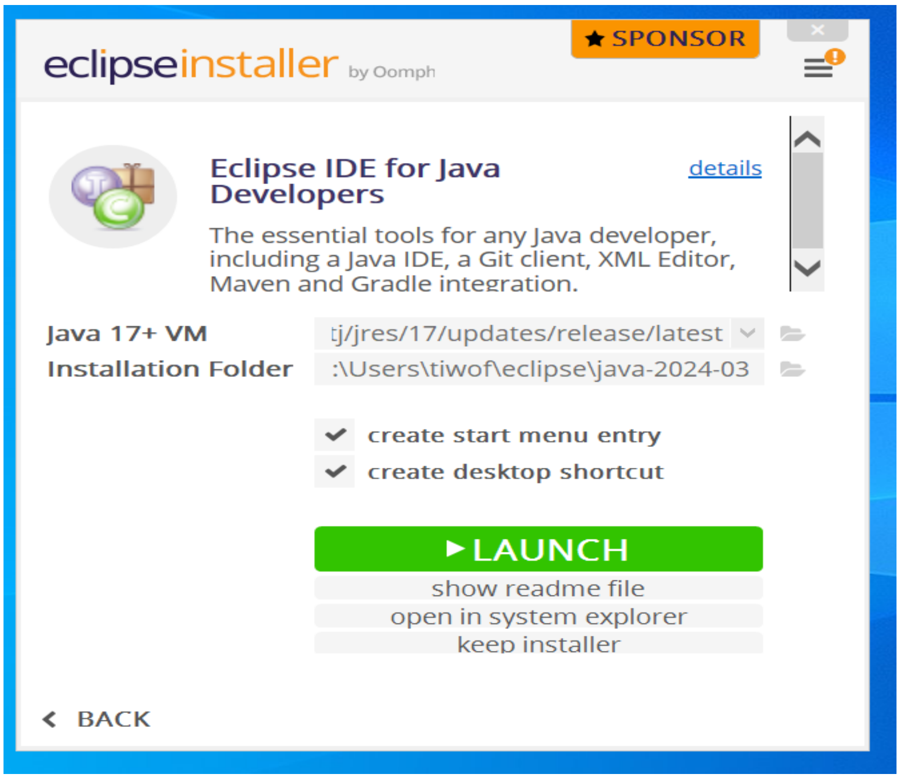
Applicable licenses will be discovered and prompted later in the installation process.

Eclipse Foundation Software User Agreement Versions

There are currently three versions of the Eclipse Foundation Software User Agreement. You may review them and accept them now, or wait until you are prompted to review and accept them later.

- Eclipse Foundation Software User Agreement Version 2.0
- Eclipse Foundation Software User Agreement Version 1.1
- Eclipse Foundation Software User Agreement Version 1.0

After the installation is completed just click launch.



Eclipse IDE for Java Developers

The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Maven and Gradle integration.

Java 17+ VM: tj/jres/17/updates/release/latest
Installation Folder: :\Users\tiwof\eclipse\java-2024-03

✓ create start menu entry
✓ create desktop shortcut

- **lease name 10 advantages of an IDE compared to a simple text editor.**

- Programming Features: IDEs come with more programming features than simple text editors. These features include auto-completion of keywords and inbuilt function names, syntax highlighting, and code navigation tools.

- Language Support: IDEs support a wide range of programming languages. Some big-name IDEs support multiple languages, while smaller ones can be useful for niche areas.
- Productivity: IDEs enhance productivity by providing a cohesive environment that integrates specialized tools. These tools may include a text editor, code autocomplete function, build procedures, compiler, linker, and debugger.
- Code Execution: IDEs provide a seamless way to execute and debug code within the same environment. They often include features like a built-in compiler and debugger, making it easier to identify and fix errors.
- Integrated Tools: IDEs offer integrated tools for version control, code refactoring, and project management. These tools streamline the development process and improve collaboration among team members.
- Error Detection: IDEs can detect errors in real-time, providing immediate feedback to the developer. This helps catch mistakes early on and improves code quality.

- **Code Templates and Snippets**: IDEs provide code templates and snippets that can be easily inserted into the code. This saves time and reduces the chances of making syntax errors.
- **Code Navigation**: IDEs offer advanced code navigation features, such as the ability to jump to function definitions, find references, and navigate through the codebase. This makes it easier to understand and navigate complex codebases.
- **Integrated Documentation**: IDEs often provide integrated documentation, allowing developers to access relevant documentation and API references without leaving the development environment.
- **Customization**: IDEs are highly customizable, allowing developers to personalize their development environment according to their preferences. This includes customizing themes, keyboard shortcuts, and plugins.

2. Prove the Ulam function defined as
1. Start with a positive integer n.
2. If n is even, divide it by 2.
3. If n is odd, multiply it by 3 and add 1.
4. Repeat steps 2 and 3 with the resulting number.

terminates with 1 for any given positive integer n<1M.

Ulam function is also known as the Collatz Conjecture, also known as the 3n+1 problem. It states that for any positive integer n, the sequence obtained by repeatedly applying the above steps will eventually reach the value 1.

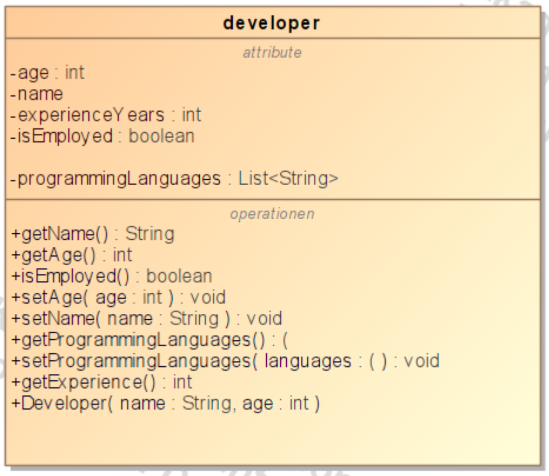
Although the Collatz Conjecture has been extensively tested for a wide range of positive integers, including those less than 1 million, a formal proof of this conjecture is still an open problem in mathematics. Despite its simplicity, the Collatz Conjecture has proven to be a challenging problem, and no counterexample has been found to disprove it.

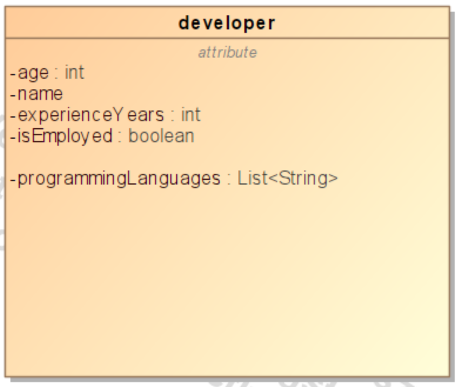
To demonstrate the complexity of this problem, let's consider an example. We'll start with the number 6:

1. 6 is even, so we divide it by 2 and get 3.
2. 3 is odd, so we multiply it by 3 and add 1, yielding 10.
3. 10 is even, so we divide it by 2 and get 5.
4. 5 is odd, so we multiply it by 3 and add 1, resulting in 16.
5. 16 is even, so we divide it by 2 and get 8.
6. 8 is even, so we divide it by 2 and get 4.
7. 4 is even, so we divide it by 2 and get 2.
8. 2 is even, so we divide it by 2 and get 1.

3. Create a github.com account and commit your software to a new repository.

5. Draw a UML class diagram for a class developer with at least 5 suitable attributes and 5 methods. Draw an object diagram with corresponding values for yourself. Implement this class in Java with integrity checks in the setter methods.


developer
attribute
-age : int
-name
-experienceYears : int
-isEmployed : boolean
-programmingLanguages : List<String>
operationen
+getName() : String
+getAge() : int
+isEmployed() : boolean
+setAge(age : int) : void
+setName(name : String) : void
+getProgrammingLanguages() : (
+setProgrammingLanguages(languages : ( ) : void
+getExperience() : int
+Developer(name : String, age : int)



6. Define the term software design. Explain how this differs from software analysis.

Software Design refers to the process of creating a plan or blueprint for a software system before coding begins. It involves making decisions about the overall structure, architecture, and behavior of the software. The design phase focuses on how the software will be implemented, taking into account factors such as functionality, performance, maintainability, and user experience. During software design, various design models and techniques are used to represent the system's structure and behavior, such as flowcharts, data-flow diagrams, state transition diagrams, and entity relationship diagrams
https://en.wikipedia.org/wiki/Software_design

Software Analysis, on the other hand, is the process of understanding and documenting the requirements and constraints of the software system. It involves studying the problem domain, identifying user needs, and defining the functional and non-functional requirements of the software. Software analysis is typically performed before the design phase and provides the foundation for the design process software analysis is concerned with understanding and documenting the requirements of the software system, while software design involves creating a plan or blueprint for implementing those requirements. Analysis comes before design and provides the necessary information and understanding to guide the design process.

7. Explain why a software design is necessary for a software project. Can you think of a project without this step? What could be the consequences?

Software design is essential for a project as it organizes the system's structure, defines functionality and performance, ensures maintainability and extensibility, aids in error detection, facilitates collaboration, mitigates risks, and promotes an efficient development process. Skipping this step can lead to errors, poor performance, maintenance difficulties, communication issues, and delays.

It is important to note that skipping the software design step in a project can have consequences. Without proper design, the development process may lack structure and clarity, leading to a higher chance of errors, poor performance, and difficulties in maintenance and future enhancements. Additionally, the lack of a design can hinder effective communication and collaboration among team members, resulting in misunderstandings and delays.

8. Are the design activities of architectural design, database design, user interface design and component design independent or interdependent. Using an example, explain why.

The design activities of architectural design, database design, user interface design, and component design are interdependent. They influence and impact each other in the software design process. For example, the architectural design influences the design of other components like the user interface and database. Similarly, the database design affects the user interface design, as specific data needs to be stored and retrieved. The user interface design aligns with the overall system structure and functionality, which is determined by the architectural design. Component design also relies on the architectural and database designs to ensure compatibility and integration.