|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Zeit | Donnerstag | Freitag | WE | Montag | Dienstag | Mittwoch | Donnerstag | Freitag |
| 09.30 – 10.30 | **Java Basic** | Grundlagen sowie Vor- und Nachteile Java |  | Describe the java.lang package | Parsen über [Wrapper](file:///C:\Users\DLabo\IdeaProjects\sic-juc-3a\Unterricht\src\_20241129WrapperCasten\DungeonCrawl%20Tag%205%203a%20JuC.pdf)-Klassen und Strings + (Auto)-(Un-)Boxing | Wann wird ein Compiler- oder Laufzeitfehler eintreten? | Welche sind gängige (un-) checked Exceptions und wo? | Vorbereitung auf die Prüfungssituation |
| 10.45 – 11.50 | **Planung** | „ |  | Sichtbarkeitsbereich und Lebensspanne von Variablen | Weiter mit Wrappern | **Quiz** | Weiter mit Exceptions | Vorbereitung auf die Prüfungssituation **Quiz** |
| 12.00 - 12.45 | Vertiefen (Post/Pre)-In/Dekrementieren | „ |  | Puffer **Ich mache Englisch4IT** | Prüfe Lernziel Bereite SBLZ vor | Pause (etwas verspätet) | Weiter mit Exceptions | **Quiz** |
| Pause |  |  |  |  |  |  |  |  |
| 13.30 – 14.30 | **TU-Quizzen** | Reste, Reflexion und Anpassung Lernplan |  | **TU-Quizzen** | Programmieren Dungeon Crawl agility, balancing | **TU** | **TU** | Math.random() vs. java.lang.util.Random (Erstes finde ich schwerer) |
| 14.45 – 15.45 | Einsatzmöglichkeiten  Wo ist es überall? (Videovortrag?) |  | Programmieren Dungeon Crawl agility, balancing | Weiter mit Randoms |
| 16.00 -17.00 | JDK, JRE, JVM, usw. Flussdiagramme und Theorie vertiefen + Auswertung |  | Programmieren Dungeon Crawl agility, balancing | Kleinkram, Fragen (RunTimeException, BattleShip-Matrix) |
| Zu Hause | - |  | Legende | erfolgreich erledigt | in Arbeit | nicht so wichtig | Ab jetzt nur noch **Quiz** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Was ist Java?** | **Java Basic** | **Basic Java Elements** | **Working with Data Types** | **Java Operators** |
| **String Class** | **Random and Math Class** | **Decision Statements** | **Looping Statements** | **Debugging and Exception Handling** |
| **Arrays und ArrayList** | **Classes and Constructor** | **Java Methods** | **Quizfragen** | **Pause** |
| **Programmieren** |  |  |  |  |
|  |  |  |  |  |

**Aufgabe**

Erstelle deinen individuellen Lernplan. Achte darauf die Dinge auf jeden Fall zu wiederholen, welche dir besonders schwerfallen.

1. Fülle deine Selbsteinschätzung erneut aus. Entweder auf Git oder kopiere dir die OneNote-Seite: [Selbsteinschätzung](onenote:https://smartindustrycampus.sharepoint.com/sites/JuCOnline3a-IT-FachkraftfrJavaundCEntwicklungIHKKopie/SiteAssets/JuC%20Online%203a%20-%20IT-Fachkraft%20für%20Java%20und%20C++%20Entwicklung%20(IHK)-Notizbuch/_Collaboration%20Space/Selbsteinschätzung.one#section-id={7721603B-AC9E-4D9E-A4FC-556627DF657E}&end)  ([Webansicht](https://smartindustrycampus.sharepoint.com/sites/JuCOnline3a-IT-FachkraftfrJavaundCEntwicklungIHKKopie/_layouts/OneNote.aspx?id=%2Fsites%2FJuCOnline3a-IT-FachkraftfrJavaundCEntwicklungIHKKopie%2FSiteAssets%2FJuC%20Online%203a%20-%20IT-Fachkraft%20f%C3%BCr%20Java%20und%20C%2B%2B%20Entwicklung%20%28IHK%29-Notizbuch&wd=target%28_Collaboration%20Space%2FSelbsteinsch%C3%A4tzung.one%7C7721603B-AC9E-4D9E-A4FC-556627DF657E%2F%29)) Lade anschließend die ausgefüllte Version ebenfalls wieder hoch.
2. Mache dir eine Liste von Themen, welche du unbedingt wiederholen möchtest.
3. Fülle deinen Wochenplan mit den entsprechenden Oberthemen.

**Achtung**:

* + Du darfst **maximal drei** Blöcke als „Pause“ markieren.
  + Du solltest **mindestens drei** Blöcke Quizfragen beantworten.
  + Und **mindestens vier** verschiedene Themen wiederholen.

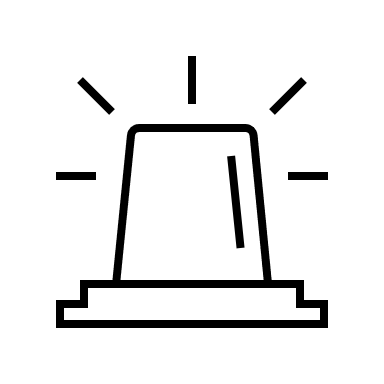
1. Lade deinen persönlichen Lernplan bei Git hoch! Ordner: \_20241216Bis20WeekPrep1811. Lernplan (In Unterricht)

**Was kannst du nutzen zu Lernen?**

* youTube
* <https://www.geeksforgeeks.org/>
* <https://www.w3schools.com/java/default.asp>
* <https://www.javatpoint.com/features-of-java>
* <https://openbook.rheinwerk-verlag.de/javainsel/>
* <https://wwwlehre.dhbw-stuttgart.de/~kfg/java/java.pdf>
* Deshmukh
* Alte Aufgaben und Folien
* Dozentin und Tutoren

**Verschiedene Aufgabensammlungen:**

* <https://www.thu.de/de/org/I/vorkurse/Documents/java_aufgaben.pdf>
* <https://tutego.de/javabuch/aufgaben/>
* <https://wiki.freitagsrunde.org/Javakurs/%C3%9Cbungsaufgaben>
* <https://info-wsf.de/uebungsaufgaben-erste-aufgaben-in-java/>
* <https://www.programmieraufgaben.ch/uploads/java.pdf>



**Wichtig**: Es muss jederzeit nachweisbar sein, was ihr gemacht habt!

|  |  |  |  |
| --- | --- | --- | --- |
| Topic | Detail | Wiederholen | Notizen |
| What is Java? |  |  | **1-2h (Gerne auch Youtube-Videos)** |
|  | Describe the features of Java |  | **Grundlagen sowie Vor- und Nachteile** |
|  | Describe the real-world applications of Java |  | **Wofür wird es nochmal alles eingesetzt?** |
| Java Basic |  |  | **1h** |
|  | Describe the Java Development Kit (JDK) and the Java Runtime Environment (JRE) |  | **Nochmal versuchen Flussdiagramme und Theorie zu verstehen** |
|  | Describe the components of a basic Java program |  | - |
|  | Describe the components of object-oriented programming |  | - |
|  | Compile and execute a Java program |  | - |
| Basic Java Elements |  |  | - |
|  | Identify the conventions to be followed in a Java program |  | - |
|  | Use Java reserved words |  | - |
|  | Use single-line and multi-line comments in Java programs |  | - |
|  | Import other Java packages to make them accessible in your code |  | - |
|  | Describe the java.lang package |  | **Was ist außer Math und System noch alles drin (und nicht)?** |
| Working with Data Types |  |  | - |
|  | Declare and initialize variables including a variable using final |  | - |
|  | Cast a value from one data type to another including automatic and manual promotion |  | **Parsen über Wrapper-Klassen und Strings (1-2h)** |
|  | Declare and initialize a String variable |  | - |
| Java Operators |  |  | - |
|  | Use basic arithmetic operators to manipulate data including +, -, \*, /, and % |  | - |
| erledigt | Use the increment and decrement operators |  | **Post/Pre-In/Dekrementieren Unterschiede und Auswirkung** |
|  | Use relational operators including ==, !=, >, >=, <, and <= |  | - |
|  | Use arithmetic assignment operators |  | - |
|  | Use conditional operators including &&, ||, and ? |  | - |
|  | Describe the operator [precedence](https://www.google.com/search?sca_esv=2a01fdbcda2f6ea4&sxsrf=ADLYWILVdMzE6mhAbZOO9NQAeDDqydlIuw:1736416613512&q=operator+precedence+java&udm=2&fbs=AEQNm0A6bwEop21ehxKWq5cj-cHa02QUie7apaStVTrDAEoT1CkRGSL-1wA3X2bR5dRYtRGv3dh0WX48pQ0OijG3Ir_Ily36WNjIM66TUeQQm6v5pCxPr2gtqfjkC7ffv6Tr7pov6Kj4r20q4qdHCSHuZ8l9l_oCqEwoxOcaGtTQ9oNU0Tr95ug&sa=X&ved=2ahUKEwjcwLj7r-iKAxVnxQIHHVdXFJcQtKgLegQIFhAB&biw=1920&bih=911&dpr=1#vhid=j3cEZAbtTLTASM&vssid=mosaic) and use of parenthesis |  | **Vorrang von Operatoren mit und ohne Klammern -Video dazu?**  **Implizit eigentlich schon verstanden, nicht verschlimmbessern!** |
| String Class |  |  | - |
|  | Develop code that uses methods from the String class |  | - |
|  | Format Strings using escape sequences including %d, %n, and %s |  | - |
| Random und Math Class |  |  | - |
|  | Use the Random class |  | - |
|  | **Use the Math class** |  | **Was kann die noch so alles?** |
| Decision Statements |  |  | - |
|  | Use the decision making statement  (if-then and if-then-else) |  | - |
|  | Use the switch statement |  | - |
|  | Compare how == differs between primitives and objects |  | - |
|  | Compare two String objects by using the compareTo and equals methods |  | - |
| Looping Statements |  |  | - |
|  | Describe looping statements |  | - |
|  | Use a for loop including an enhanced for loop |  | - |
|  | Use a while loop |  | - |
|  | Use a do- while loop |  | - |
|  | Compare and contrast the for, while, and do-while loops |  | - |
|  | Develop code that uses break and continue statements |  | - |
| Debugging & Exception Handling |  |  | - |
|  | Identify syntax and logic errors |  | Wann wird ein Compiler- oder Laufzeitfehler eintreten? |
|  | Use exception handling |  | Welche sind gängige (un-)checked Exceptions und wo? |
|  | Handle common exceptions thrown |  | - |
|  | Use try and catch blocks |  | - |
| Arrays und ArrayList |  |  | - |
|  | Use a one-dimensional array |  | - |
|  | Create and manipulate an ArrayList |  | - |
|  | Traverse the elements of an ArrayList by using iterators and loops including the enhanced for loop |  | Wenn noch Zeit ist (nicht für die Prüfung momentan nötig) schauen, wie das Iterator-Interface funktioniert. Es reicht sonst zu wissen, was es in der Aufgabe tut. |
|  | Compare an array and an ArrayList |  | - |
| Classes and Constructor |  |  | - |
|  | Create a new class including a main method |  | - |
|  | Use the private modifier |  | - |
|  | Describe the relationship between an object and its members |  | - |
|  | Describe the difference between a class variable, an instance variable, and a local variable |  | **Sichtbarkeitsbereich und Lebensspanne von Variablen** |
|  | Develop code that creates an object's default constructor and modifies the object's fields |  | - |
|  | Use constructors with and without parameters |  | - |
|  | Develop code that overloads constructors |  | - |
| Java Methods |  |  | - |
|  | Describe and create a method |  | - |
|  | Create and use accessor and mutator methods |  | - |
|  | Create overloaded methods |  | - |
|  | Describe a static method and demonstrate its use within a program |  | - |