|  |  |  |
| --- | --- | --- |
| Emne |  | Kommentarer og spørgsmål |
| 01: Objekter |  | Hvad er en klasse, et objekt, en constructor, gettere/settere, toString(), access modifiers (private/public/protected), hvordan instantierer man et objekt? |
| 02: ArrayList |  | Hvordan erklærer og instantierer man en ArrayList? Simple typer og egne typer? Hvad er det smarte ved en arrayList i forhold til et array? Hvilke metoder anvender vi på arraylister? Det er bl.a. add(), size(), contains(), indexOf(), remove(), get(). Hvad gør det? |
| 03: Arrays |  | Hvad bruger man arrays til og hvad kan de indeholde? Hvordan gemmer man værdier og objekter i et array? |
| 04: Scanner |  | Indlæsning fra tastatur. Hvordan kan vi sørge for at fange fejlsituationer under indlæsning af f.eks. doubles eller integers fra tastaturet? Hint: try-catch. |
| 05: Strings |  | Det bruger vi overalt. Fokuser på hvilke metoder, der hører til String klassen. F.eks. length(), charAt(), toLowerCase(). |
| 06: Metoder |  | Vi bruger metoder overalt fra og med uge 38. Tænk over hvorfor det er smart at bruge metoder og vær klar på at forklare hvad en metode-signatur består af. Hvad vil det sige at assigne en returværdi til en variabel? (en:assignment, da:tildeling) |
| 07: Logik |  | if-then-else, switch-case-break, &&, ||, booleans Hvornår kan man vælge en switch-case istedet for et if-statement? Hvad er forskellen? |
| 08: Loops |  | for- og while løkker og for-each i forbindelse med ArrayLists. Sørg for at du er helt hjemme i syntaksen så du relativt hurtigt kan gennemløbe et array. Det skal trænes. |
| 09: Variabeltyper |  | int, Integer, String, double, boolean. Du skal kende de mest gængse datatyper, og hvad de bruges til. Det er godt hvis du kan forklare hvorfor man ikke kan sammenligne to String værdier med ==, som man kan med de primitive datatyper. |
| 10: Debugging |  | Vis hvordan man kan debugge et stykke kode og løbende se metodekald og indhold af diverse variabler |
| 11: Exceptions |  | Hvad er try-catch og exceptions? Hvad er throw / throws? Hvornår vil man typisk håndtere en fejl i en catch og hvornår smider man fejlen videre? |
| 12: Filer |  | Vis f.eks. hvordan I har indlæst fra filer i Marios - og gemt ordrer. Hvad kan gå galt når man indlæser fra en fil - og hvordan håndterer man det? Hint: try-catch |
| 13: JUnit tests |  | Hvad er meningen med [Unit-test](https://youtu.be/rKcuENQ_uXQ%20https:/youtu.be/q5C9i3pb3EM%20https:/youtu.be/LEslVYkgSLw)s? Hvad er TDD? Hvad gør assertEquals metoden? Hvad kan man gøre hvis man gerne vil teste om en void metode har haft den forventede effekt? |
| 14: JDBC |  | Hvad er JDBC? Giv eksempler på håndtering af data fra en database i applikationslaget. |
| 15: Nedarvning |  | Hvordan virker nedarvning. Hvad er en superklasse? Hvordan ser en constructor ud for en sub-klasse og hvad betyder "super". Hvordan anvender man nedarvning (Hint: extends). Hvad er formålet med nedarvning? (tænk på D-R-Y princippet) |
| 16: Interfaces |  | Hvad er et interface (Hint: en kontrakt) og hvordan bruger man det (Hint: implements)? Giv eksempler hvor vi bruger nogle af Javas interfaces ([Comparable](https://www.youtube.com/watch?x=34&v=ZBH2rDKluu0&feature=emb_logo)) |
| 17: **Databaser og SQL** |  | Hvad er en database, tabeller, felter/kolonner, primær nøgler, fremmed-nøgler, et EER-diagram, CRUD, MySql, Workbench |
| 18: **UML** |  | Hvad er et Use-Case diagram? Hvad er en Domæne-model? Hvad er et Klasse-diagram?  Hvornår er de nyttige i udviklingsprocessen? |
| 19: **Datastrukturer** |  | Lister, Sets og Maps. Hvad er en datatypes 'naturlige orden'? Hvad er nødvendigt hvis man vil sortere en ArrayList af objekter som du selv har skrevet klassen til? (hint: brug comparable eller comparator interfacet)  https://www.javatpoint.com/data-structure-introduction |