

Individuell inlämningsuppgift - översikt

- Två delar på denna uppgift
 1. Beskrivning av det du har bidragit med under grupparbetet
 2. Svara på ett antal frågor
- Till sist ska du lämna in ett markdown-dokument med denna information på Totara

Individuell inlämningsuppgift - tillvägagångssätt

- I IntelliJ eller Visual Studio Code - skapa en markdown-fil
- I IntelliJ skapar man en ny fil och döper den till någonting.md

Individuell inlämningsuppgift - tillvägagångssätt

- I markdown-filer kan man skriva kod i speciella kod-block
- Kod-blocken påbörjas med tre 'back-ticks': `` `
- Direkt efter dessa tre symboler skriver man ordet java med små bokstäver
- Därefter trycker man retur
- Sedan kan man skriva flera rader kod precis som i vanlig kod
- Kodblocket avslutas till sist med tre 'back-ticks': `` `
- Vanligtvis på en egen rad efter all kod

Individuell inlämningsuppgift - tillvägagångssätt

- I markdown-filen ska du svara på ett antal frågor och även ge kodexempel, kodexempel ger ni på det sätt som beskrevs på förra sidan.
- Varje fråga är numrerad. I din markdown-fil anger du med numret vilken fråga det handlar om
- En rubrik i markdown skrivs så här: # Din rubrik
- Innan ditt svar skriver du frågans nummer som rubrik
- Ni får använda hjälpmedel för att besvara frågorna, men inte hjälpa varandra

Individuell inlämningsuppgift - beskrivning av vad du har gjort

- Allra först i filen ska du dock beskriva vad du har bidragit med under grupparbetet.
- Vilken kod har du bidragit med?
- Beskriv utförligt vad just din kod gör, ge kodexempel med kodblock och förklara vad koden gör

Betyg G: frågor och kodexempel

- Fråga 1: Ge exempel på en metod i din grupps webbapplikation. Skriv metoden i ett kodblock.
- Fråga 2: Förklara följande: vad innebär ordet public/private i metodssignaturen?
- Fråga 3: I metodssignaturen följer därefter ett annat ord ('void' eller namnet på en typ eller klass). Vad innebär det om det står 'void'? Vad innebär det om det står en typ eller klass?
- Fråga 4: Efter typen/klassen skriver man metodens namn. Vilken namnkonvention finns för metoder i Java?
- Fråga 5: Efter metodens namn skrivs ibland bara parantes-tecken: (). Vad innebär det om det inte står något mellan paranteserna?

Betyg G: frågor och kodexempel forts

- Fråga 6: Vad kallas det som skickas med till metoden mellan paranteserna? Det finns iallafall två namn för vad det kallas.
- Fråga 7: Här följer ett metod-exempel:

```
public void add(int a, int b){  
  
}
```

- Hur ska du skriva för att returnera summan av a och b i denna metod? Kopiera metoden och skriv dit det som saknas. Använd kodblock.
- Fråga 8: Hur ser det ut när man anropar denna metod från en annan metod? Skapa ett kodblock och visa hur det kan se ut när denna metod anropas.

Betyg G: frågor och kodexempel forts

- Fråga 9: Se denna klass:

```
public class HakansClass{  
  
}
```

- Skapa ett kodblock och kopiera in min klass. Lägg till en tom konstruktor (utan argument) i ditt svar.
- Fråga 10: Vad är det för skillnad på objekt och primitiva typer i Java? Svara med ett kodblock där du visar 1) skapandet av en primitiv typ med ett värde, 2) skapandet av ett objekt (en instans av en klass).

Betyg G: frågor och kodexempel forts

- Fråga 11: Med Spring används speciella @Bean-annotationer. Det är metoder som annoteras på det sättet. Konventionen i Spring är att metoder annoterade med @Bean namnges med den typ som returneras, ex:

```
@Bean
public HakansClass hakansClass(){
    return new HakansClass();
}
```

Men hur är annars namnkonventionen i Java när det gäller metoder? Vad brukar man börja metodnamnet med (tips: en ordklass)?

Betyg VG (utöver frågorna för G)

- Fråga 12: Vad innebär annotationen @Override och när används den?
- Fråga 13: När en klass ärver av en annan klass använder man keywordet extends, ex:

```
public class MyCustomAuthenticationProver extends DaoAuthenticationProvider{  
    public MyCustomAuthenticationProver(){  
  
    }  
  
}
```

Klassen DaoAuthenticationProvider (som finns i Spring) har denna metod:

```
authenticate(Authentication auth)
```

- kan jag kalla på den metoden i subklassen i MyCustomAuthenticationProvider?
Hur skriver jag för att kalla på den metoden, som alltså är deklarerad i superklassen?

Betyg VG forts

- Fråga 14: Vilka uppgifter har PasswordEncoder-@Bean? Dels vid registrering, dels vid inloggning?
- Fråga 15: Logback är default-implementationen för loggning i Spring. Skapa koden till en konfigurationsfil för logback.xml. Filen ska innehålla: en appender som loggar till fil, en ROLLING file-appender (som genererar en loggfil per dag), en logger för baspaketet se.sprinto.hakan.securityapp och som loggar till ROLLING file, samt en root-logger som loggar till den andra filen. Använd kodblock, men detta är xml så du skriver `` `xml istället för java.
- Fråga 16: Vad innebär HTTP-koderna: 200, 401, 403 och 404?