

# Kunskapskontroll – Python, NBI

Denna kunskapskontroll består av två delar.

1. Den första delen består av teoretiska frågor och python-övningar
2. Den andra delen är kopplat till ett dataset som du kommer analysera med hjälp av Python. Som du kommer märka i AI kursen efter sommaren är det något vi kommer ha stor nytta av när vi praktiskt tillämpar AI.  
För analysen du gör ska du även skapa en Streamlit applikation där du presenterar det du vill presentera.

Hela kunskapskontrollen, bestående av de två delarna, lämnas in i en jupyter notebook i LearnPoint samt ett separat skript som innehåller Streamlit applikationen. Betygen IG/G/VG kan erhållas. Se kursplanen för betygskriterier.

I slutet av din jupyter notebook ska du besvara följande självutvärdering:

1. Vad har varit roligast i kunskapskontrollen?
2. Vilket betyg anser du att du ska ha och varför?
3. Vad har varit mest utmanande i arbetet och hur har du hanterat det?

## Del 1 – Teoretiska frågor och Python-övningar

1. Vad är skillnaden på en *tuple* och *list* i Python? Är den ena datatypen bättre än den andra?
2. Vad är en funktion för något? Varför kan funktioner vara användbara?
3. Förklara vad nedanstående begrepp, kopplat till klasser, betyder:
  - a) Instans
  - b) Attribut
  - c) Metod
4. Vad är Streamlit för något?
5. I Python ska du nu:
  - a) Skapa en klass som heter BankAccount. Klassen skall ha attributet "account\_holder" som visar kontoinnehavarens namn samt attributet "balance" som visar kontoinnehavarens balans. Klassen skall ha metoden "deposit()" för att kunna sätta in pengar på kontot samt metoden "withdraw()" för att kunna ta ut pengar från kontot. Om bankinnehavaren försöker ta ut mer pengar än vad som finns på kontot skall meddelandet "Too low balance" printas ut.
  - b) Skapa en instans av klassen och testa så klassen funkar så som du förväntar dig. Du kan till exempel prova printa ut attributen, sätta in pengar och ta ut pengar.
6. På intervjuer brukar ibland kodtest genomföras. Här är ett exempel på en fråga som hade kunnat dyka upp på ett kodtest.  
Skapa en Python funktion som beräknar antalet vokaler i en sträng. Vokalerna är: A, E, I, O, U, Y, Å, Ä och Ö.

Exempelvis: `vowel_checker('hjärna')` ska returnera "2" eftersom ordet "hjärna" har de två vokaler "ä" och "a".

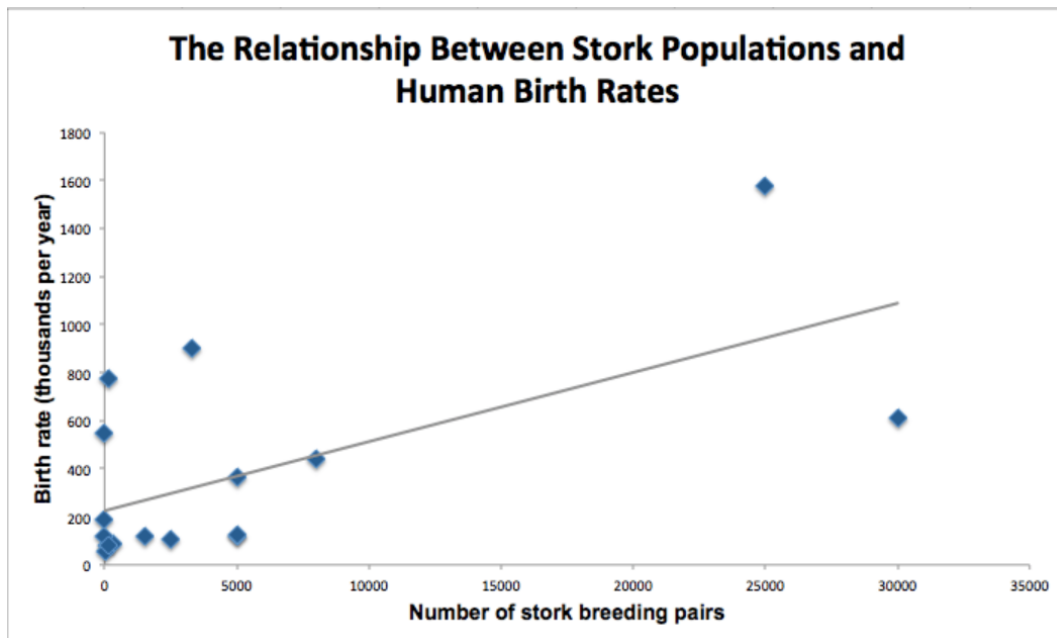
7. På intervjuer brukar ibland kodtest genomföras. Här är ett exempel på en fråga som hade kunnat dyka upp på ett kodtest.

Skapa en funktion som tar två listor som argument och returnerar de gemensamma elementen i listorna i en ny lista.

Exempelvis:

`common_elements([4, 'apple', 10, 'hi', 3], ([23 'apple', 5, 9, 3])` ska returnera svaret `['apple', 3]`.

8. Många barn tror att de blev till genom att en stork kom och lämnade dem till deras föräldrar. För att undersöka det kan vi kolla på data och det finns en graf enligt figuren nedan. Det verkar alltså finnas ett samband mellan antalet storkar och barnafödelsel!? Stämmer påståendet? Använd begreppet "kausalitet" i ditt svar.



9. Kim påstår att "medelvärde" är ett bättre mått än "medianvärdet". Håller du med Kalle?
10. Vad används cirkeldiagram för? Ge ett exempel på vad ett företag som Spotify (eller något annat företag som du själv väljer) hade kunnat använda cirkeldiagram för.
11. Vad används linjediagram för? Ge ett exempel på vad ett företag som Spotify (eller något annat företag som du själv väljer) hade kunnat använda linjediagram för.
12. Vad används lådagram för?

## Del 2 – Dataanalys i Python

”Business caset” är att du arbetar på en IT avdelning på ett företag som heter ”Guldfynd”. Företaget har butiker i hela Norden. Företaget är specialiserat på smycken innehållande guld och silver. För att utveckla sin verksamhet överväger de att genomföra investeringar för att även erbjuda produkter med diamanter bland sitt sortiment. Därför har de fått tillgång till ett dataset kopplat till just diamanter. Du är ansvarig för att genomföra analysen i Python. Din analys behöver innehålla olika visualiseringar. Vilka lärdomar, insikter och rekommendationer kan du förmedla?

Datasetet som ska analyseras heter ”diamonds” och är lagrat i en CSV fil.

- Bonusmaterial: Den läsare som vill lära sig Excel, kan se kapitel 1-6 och kapitel 8 i följande video:  
<https://www.youtube.com/watch?v=4UMLFC1SoHM&list=PLgzaMbMPEHEx2aR9-EXfD6psvezSMcHJ6&index=1&t=4s>  
Notera, Excel videon är inget som ingår i kunskapskontrollen utan är endast för de som vill lära sig Excel. Excel är generellt sett något som anses vara ”allmänbildning” inom IT.

Utforma din ”jupyter notebook” som en ”data story”. ”Data storytelling” kan definieras enligt följande: *”Data storytelling is the concept of building a compelling narrative based on complex data and analytics that help tell your story and influence and inform a particular audience.”*

- I din kod ska du först ha en kort bakgrund som förklarar vad som görs i ”jupyter notebooken”. Här kan du t.ex. presentera vad syftet är.
- Därefter gör du en analys av datan och förklarar vad som kan ses från analysen. Använd dig av beräkningar, visualiseringar och skriv även vanlig text för att förklara insikter och lärdomar.
- Slutligen presenterar du huvudinsikterna och vilka rekommendationer du ger i en ”executive summary”. Den ska vara kort och innehålla det viktigaste. ”Executive summaries” är exempelvis något chefer i företag läser.

Notera, när ni gör er analys är ni fria att specificera olika antaganden eller vilka infallsvinklar ni vill ha på er analys.

För analysen du gör ska du även skapa en Streamlit applikation där du presenterar det du vill presentera. Notera, du ska alltså inte inkludera allt från din jupyter notebook utan du behöver göra avvägning kring vad du vill inkludera i din applikation. Den som har tid kan även försöka ”deploya” applikationen vilket kan göras gratis via Streamlit, läs dokumentationen.

De som satsar på VG behöver särskilt tänka på att:

1. Skriva bra och tydlig Python kod.
2. Göra en tydlig och konsistent ”data storytelling” där läsaren av koden kan följa med i progressionen och tankegången. Analysen ska även vara nyanserad där t.ex. för- och nackdelar förklaras. Varva därför kod, visualiseringar och förklaringar som är skriven i vanlig text. Undvik att göra extremt mycket beräkningar och visualiseringar utan ett tydligt syfte.
3. Skapa en bra Streamlit applikation.