|  |  |
| --- | --- |
| **Oppgave 1**x = 3while x < 10: x += 1print(x) | **Løsning**10. Tegn gjerne opp hvordan ting skjer, eventuelt bruk pythontutor.  |

|  |  |
| --- | --- |
| **Oppgave 2**Skriv et program med en variabel a med verdien 1 og en while-løkke. For hver runde i while-løkken skal variabelen a plusses med 2. Løkken skal være ferdig når verdien av a er større enn 100. Skriv så ut verdien av a. Hvilken verdi blir skrevet ut? | **Løsning**101Siste gang løkken gjennomføres er a = 99, inni løkken økes a med 2 (får verdi 101), så sjekkes det igjen om a < 100, det evalueres til False (siden a er 101). Dermed er løkken ferdig, og a skrives ut.  |

|  |  |
| --- | --- |
| **Oppgave 3**Hva tror du skrives ut på skjermen av programmet under? Diskuter med gruppen. tekst = ["hadet", "på", "badet", "din", "gamle", "sjokolade"]indeks = 0 while indeks < len(tekst): print(tekst[indeks]) indeks += 2 | **Løsning**> hadet> badet> gamleNB! Indekseringen øker med 2 for hver runde.  |

|  |  |
| --- | --- |
| **Oppgave 4**Skriv et program med en prosedyre printHei, prosedyren skal skrive ut teksten “Hei” til skjermen. Definer en variabel a med en verdi 5 og lag en while-løkke, løkken skal kalle prosedyren printHei. Bruk variabelen a slik at løkken er ferdig etter du har kalt print\_hei 5 ganger.Skriv prosedyren printHei. Deretter løs oppgaven både ved hjelp av while-løkke og for-løkke. | **Løsning**def printHei() : print("Hei")i = 0a = 5while i < a : printHei() i += 1for i in range(a) :  printHei() |

|  |  |
| --- | --- |
| **Oppgave 5**Skriv et program som inneholder en while-løkke. Inne i løkken skal bruker bes om å taste et tall. Dersom tallet ikke er 10, skal løkken fortsette å be om tall. Når bruker taster inn tallet 10, skal brukeren få beskjeden “Du har tastet inn tallet 10. Programmet avsluttes …” før programmet så avslutter.*Utfordring:* summer sammen alle tallene brukeren skriver inn frem til 10 blir tastet. Når 10 tastes inn skrives summen ut. Dvs. hvis bruker taster sekvensen 1 - 3 - 4 - 10, skrives det ut “Sum er: 8” før programmet avsluttes.  | **Løsning**tall = 0sum = 0while tall != 10 : sum += tall tekst = input("skriv tall ") tall = float(tekst)print("Sum er: ", sum) |

|  |  |
| --- | --- |
| **Oppgave 6**Hva skrives ut på skjermen av programmer under? Her kan det være nyttig å skrive ned verdiene av a og b for hver iterasjon av løkken. a = 10 b = 1  while a > 0: b = b \* 2 a = a - b print("a =", a)print("b =", b) | **Løsning**a = -4b = 8 |

|  |  |
| --- | --- |
| **Oppgave 7**Gitt mengden (!NB ikke liste) : tall\_rekke = {3, 6, 8, 1} Legg sammen alle tallene i mengden og skriv ut summen. | **Løsning**tall\_rekke = {3, 6, 8, 1}for tall in tall\_rekke : sum += tallprint(sum) |

|  |  |
| --- | --- |
| **Oppgave 8**Skriv en funksjon “storst\_av\_to” som tar imot to tall som parametre og skriver ut verdien til det største tallet. | **Løsning**def storst\_av\_to(tall1, tall2) : if tall1 > tall2 :  print(“Storst: “, tall1) else : print(“Storst: “, tall2) |

|  |  |
| --- | --- |
| **Oppgave 9**def skriv\_historie(forst, andre, tredje) : print(forst, “dro paa ferie med “, tredje,  “de ville dra uten “, andre, “  men “, andre, “snek seg med i  bagasjerommet..”)navn1 = “Silje”navn2 = “Ole ”navn3 = “Jakob”navneliste = [“Emilie”, “Haakon”, “Yulai”]skriv\_historie(“Kari”, “Per”, “Martin”)skriv\_historie(navneliste[0], navn3, navn1]skriv\_historie(navn2 + navn3, navneliste[1], navneliste[2]) | **Løsning**>Kari dro paa ferie med Martin de ville dra uten Per, men Per snek seg med i bagasjerommet..>Emilie dro paa ferie med Silje de ville dra uten Jakob, men Jakob snek seg med i bagasjerommet..>Ole Jakob dro paa ferie med Yulai de ville dra uten Haakon, men Haakon snek seg med i bagasjerommet. |

|  |  |
| --- | --- |
| **Oppgave 10**(Du skal nå endre på oppgave 8): Skriv en funksjon “storst\_av\_to” som tar imot to tall som parametre og returnerer det største tallet. Deretter skrives tallet ut etter funksjonskallet.  | **Løsning**def storst\_av\_to(tall1, tall2) : if tall1 > tall2 :  return tall1 else : return tall2storst = storst\_av\_to(1, 3)print(“Storst: “, storst) |