**Variabler**

**navn="Espen" # Venstresiden viser navnet på variabelen.**

**print(navn) # Lilla markerer innebygde spesifikke kommandoer i Pyton.**

**# Alt etter hashtag er ikke en del av koden.**

**Resultat:**

**Espen**

**person = "Espen" # Ved bruk av anførseltegn får man en string. Man kan bruke " " og ' '**

**print("Hei!",person) # print=Skriver ut.**

**Resultat:**

**Hei! Espen**

**Oppgave: Skriv en personlig hilsen til en i klassen.**

**person = "Espen" # En string er en samling av tekst.**

**print(f"Hei!{person}") # Bruk av f betyr at du skal flette noe sammen.**

**# Merk to måter å skrive ("Hei!",person) og (f"Hei!{person}")**

**Resultat:**

**Hei! Espen**

**pi=3.14 # vi bruker punktum for å markere forskjell på heltall og desimaltall**

**radius=11**

**volum = (4\*pi\*radius\*\*3)/3 # \* multiplikasjon, \*\* opphøyd i**

**print(volum)**

**Resultat:**

**5572.453333333334**

**pi= 3.14**

**radius=11**

**volum =4\*pi\*radius\*\*3/3**

**print(f"Volumet er {volum:.3f} liter.") # :.3f betyr kun tre desimaler**

**Resultat:**

**Volumet er 5572.453 liter**

**Oppgave: Endre på formelen slik at du får ut radius**

**pris = 499**

**pris \*=0.7 # Det samme som ny pris = gammel pris\*0.7, ny pris = 499 \* 0.7**

**print(f"Den nye prisen er {pris:.0f} kr.") # :.0f gir helt tall**

**Resultat:**

**Den nye prisen er 349 kr.**

**x=range(6) # range= gjelder for følgende område, begrenset tallmengde**

**print(x)**

**Resultat:**

**range(0,6) # range(0,6) gjelder for følgende område 0,1,2,3,4,og 5**

**x=bool(0) # Grønne kommandoer definerer hva slags type data du har.**

**print(x) # Boolean brukes for å definere om noe er sant eller usant.**

**Resultat:**

**False**

**print(10>9)**

**print(10==9)**

**print(10<9)**

**Resultat:**

**True**

**False**

**False**

**x=bool(1)**

**print(x)**

**Resultat:**

**True**

**Input**

**navn = input('Hva heter du? ') # input = pyton husker det du skriver**

**print('Hei på deg ',navn, '!') # Husk alltid komma rundt navnet på variabel eller f og klammeparentes { }**

**Resultat:**

**Hva heter du? Espen**

**Hei på deg Espen!**

**navn = input("Hva er navnet ditt? ")**

**alder = input("Hva er alderen din? ")**

**print("Hei! ", navn)**

**print("Alderen din er ", alder)**

**Resultat:**

**Hva er navnet ditt? Espen**

**Hva er alderen din? 48**

**Hei! Espen**

**Alderen din er 48**

**alder = int(input("Hvor gammel er du? ")) # int = registrerer kun heltall**

**print(f"Du er {alder} år gammel.")**

**Resultat:**

**Hvor gammel er du? 48**

**Du er 48 år gammel.**

**a = int(input('Oppgi temperaturen i Fahrenheit '))**

**b = 5/9\*(a-32)**

**print(f"Her har du temperaturen i Celsius {b} grader.")**

**Resultat:**

**Oppgi temperaturen i Fahrenheit 155**

**Her har du temperaturen i Celsius 68.33333333333334 grader.**

**Oppgave: Forandre formelen slik at jeg kun får 1 desimal.**

**Oppgave: Gjør om formelen slik at jeg får ut Fahrenheit**

**navn = input("Hva er navnet ditt?")**

**antall\_bokstaver = len(navn) # len gir antall bokstaver/tegn**

**print(antall\_bokstaver)**

**Resultat:**

**Hva er navnet ditt? Espen**

**5**

**If**

**alder = int(input('Hvor gammel er du?'))**

**if alder >18: # if setter betingelse**

**print('Du er gammel nok til å kjøpe øl.')**

**else: # if og else er knyttet sammen**

**print('Du er dessverre for ung til å kjøpe øl.')**

**Resultat:**

**Hvor gammel er du?48**

**Du er gammel nok til å kjøpe øl.**

**Hvor gammel er du?15**

**Du er dessverre for ung til å kjøpe øl**

**x = int(input("Velg et tall "))**

**if (x < 10): # tre ulike if betingelser som er uavhengig av hverandre.**

**print("Tallet er mindre enn 10")**

**if 2<x and x<10: # Du får tilbakemelding på alle if betingelsene.**

**print("Tallet er større enn 2 og mindre enn 10")**

**if x<6 or x<10:**

**print("tallet er mindre enn 6 eller større enn 10")**

**Resultat:**

**Velg et tall 5**

**Tallet er mindre enn 10**

**Tallet er større enn 2 og mindre enn 10**

**tallet er mindre enn 6 eller større enn 10**

**tall1 = int(input("Velg et tall "))**

**tall2 = int(input("Velg et tall "))**

**if tall1 > tall2: # if og elif hører sammen, dvs. at du får kun en tilbakemelding.**

**print("Tall en er størst")**

**elif tall2 > tall1: # Du kan ha så mange elif som du måtte ønske.**

**print("Tall to er størst")**

**elif tall1 == tall2: # == betyr at to enheter er identiske.**

**print("Tallene er like")**

**Resultat:**

**Velg et tall 4**

**Velg et tall 7**

**Tall to er størst**

**Velg et tall 5**

**Velg et tall 5**

**Tallene er like**

**poeng=0**

**svar = input("Hva er hovedstaden i Tyrkia? ")**

**if svar == "Ankara" or svar == "ankara": # godtar ulik skrivemåter**

**print("Det stemmer!")**

**poeng +=1 # ny poengsum = poeng +1, ny poengsum = 0+1**

**print(f"Sum poeng {poeng}")**

**else: # riktig rekkefølge er if, elif og til sist else.**

**print("Feil, riktig svar er Ankara")**

**Resultat:**

**Hva er hovedstaden i Tyrkia? ankara**

**Det stemmer!**

**Sum poeng 1**

**Oppgave: Legg til flere spørsmål.**

**a=int(input("Oppgi et heltall "))**

**b=int(input("Oppgi et heltall "))**

**rest =a%b # Modulus gir rest, eks 7%2= 7-2-2-2= 1 i rest.**

**if rest==0:**

**print("Tallet", a ,"er delelig med.", b)**

**elif rest != 0: # != betyr at to enheter ikke er identiske**

**print("Tallet", a, "er ikke delelig med.",b)**

**Resultat:**

**Oppgi et heltall 7**

**Oppgi et heltall 2**

**Tallet 7 er ikke delelig med 2.**

**Lister**

**oddetall = list(range(1,10,2)) # list betyr at du lager en hakeparentes liste**

**print(oddetall) # (1,10,2) siste tallet betyr at du kun tar annenhver**

**Resultat:**

**[1, 3, 5, 7, 9] # Ved bruk av hakeparentesliste kan du legge til i ettertid.**

**min\_liste = [3,5,3,6,8] # Nummerering etter indeks 0,1,2,3,4 eller -5,-4,-3,-2,-1**

**min\_liste.append(2) # append betyr at du legger til en navngitt hakeparentesliste**

**print(min\_liste[0]) # skriver ut nr 0 som er 3**

**print(min\_liste[-1]) # skriver ut nr -1 som er 2, fordi 2 ble lagt på bakerst i listen**

**print(min\_liste[3]) # skriver ut nr 3 som er 6**

**print(len(min\_liste)) # len betyr at du skriver ut sum antall enheter i listen**

**Resultat:**

**3**

**2**

**6**

**6**

**navn=(”Ole”,”Per”,”Pål”) # Ved bruk av parentes kan ikke listen endres på i ettertid.**

**print(navn) # En slik liste kalles for en tuple og kan ha like verdier.**

**Resultat:**

**(”Ole”,”Per”,”Pål”)**

**Salg\_pr\_mnd=[20k]\*12**

**print(Salg\_pr\_mnd)**

**Resultat:**

**[20k,20k,20k,20k,20k,20k,20k,20k,20k,20k,20k,20k]**

**For løkker**

**for x in range(5): # for løkken er begrenset av range. Løkken skal gjentas 5 ganger.**

**print("Espen") # Det som kommer etter kolon skal gjentas 5 ganger.**

**Resultat :**

**Espen**

**Espen**

**Espen**

**Espen**

**Espen**

**for hilsen in range(5): # Istedenfor x og andre symboler kan man bruke ord som gir mer fornuft.**

**print("Hei, verden")**

**Resultat:**

**Hei, verden**

**Hei, verden**

**Hei, verden**

**Hei, verden**

**Hei, verden**

**for x in range(1,101): # Skriver ut alle tall fra og med 1 til og med 100.**

**print(x)**

**Resultat:**

**1**

**2**

**3**

**4**

**5**

**6**

**7**

**8**

**.**

**.**

**.**

**93**

**94**

**95**

**96**

**97**

**98**

**99**

**100**

**for tall in range(1,10,2): # bruker ordet tall som gir mer fornuft**

**print(tall)**

**Resultat:**

**1**

**3**

**5**

**7**

**9**

**for x in “banan”: # x blir her den enkelte bokstav i banan**

**print(x)**

**Resultat:**

**b**

**a**

**n**

**a**

**n**

**s = ”Dette er en test”**

**for x in s: #Aksesserer tegnene direkte, x blir først D, deretter e osv.**

**print(x)**

**Resultat:**

**D**

**e**

**t**

**t**

**e**

**e**

**r**

**e**

**n**

**t**

**e**

**s**

**t**

**s = ”Dette er en test” # Noen ganger kan posisjonen i strengen/teksten være viktig.**

**for i in range(len(s)): #Aksesserer tegnene via indeks, i blir først 0, deretter 1 osv.**

**print(i,s[i]) # Først indeks tall deretter bokstavene etter indeks, 0=D, 1=e, 2=t osv.**

**Resultat:**

**0 D**

**1 e**

**2 t**

**3 t**

**4 e**

**5**

**6 e**

**7 r**

**8**

**9 e**

**10 n**

**11**

**12 t**

**13 e**

**14 s**

**15 t**

**s = "Dette er en test"**

**for bokstaver in s: # Merk at mellomrom også blir sett på som en enhet.**

**print(bokstaver)**

**Resultat:**

**D**

**e**

**t**

**t**

**e**

**e**

**r**

**e**

**n**

**t**

**e**

**s**

**t**

**navn = ["Espen", "Ole", "Per"] # indeks nr 0,1,2**

**for i in range(len(navn)): # len er 3, range(0,3), for løkka gjelder da for 0,1,2**

**print(f"Hei! {navn[i]}") # Fletter sammen navn etter indeks nr fra listen.**

**Resultat:**

**Hei! Espen**

**Hei! Ole**

**Hei! Per**

**tall = int(input("Velg et tall mellom 1 og 10"))**

**for x in range(1,11): # for løkka gjelder for tall nr 1 til og med 10**

**print(f"{tall}\*{x}={tall\*x}") # gjentar regnestykket for x=1 opp til x=10**

**for x in range(1,101,tall): # 1 + tall= neste tall som skal skrives ut, osv.**

**print(x)**

**Resultat:**

**Velg et tall mellom 1 og 10**

**6\*1=6**

**6\*2=12**

**6\*3=18**

**6\*4=24**

**6\*5=30**

**6\*6=36**

**6\*7=42**

**6\*8=48**

**6\*9=54**

**6\*10=60**

**1**

**7 # 1+6=7**

**13 # 7+6=13**

**19**

**25**

**31**

**37**

**43**

**49**

**55**

**61**

**67**

**73**

**79**

**85**

**91**

**97**

**måneder=["januar","februar","mars","april","mai","juni","juli","august","september","oktober","november","desember"]**

**månednummer = int(input("Hvilken månednummer vil du vite navnet på? "))**

**print(måneder[månednummer-1]) # Må trekke fra 1, siden listen følger indeks nr.**

**print(måneder)**

**print(len(måneder))**

**for i in range(len(måneder)): # i betyr at du følger indeks, derfor blir range(0,12)**

**print(i+1,måneder[i]) # Må plusse på en siden indeksen begynner på null.**

**Ellers vil første måned bli, 0 Januar.**

**Resultat:**

**Hvilken månedsnummer vil du vite navnet på? 6**

**juni**

**['januar', 'februar', 'mars', 'april', 'mai', 'juni', 'juli', 'august', 'september', 'oktober', 'november', 'desember']**

**12**

**1 januar**

**2 februar**

**3 mars**

**4 april**

**5 mai**

**6 juni**

**7 juli**

**8 august**

**9 september**

**10 oktober**

**11 november**

**12 desember**

**total=0**

**for tall in range(1,11):**

**total +=tall # total + tall = ny total**

**print(total)**

**Resultat:**

**1 # 0+1=1**

**3 # 1+2=3**

**6 # 3+3=6**

**10**

**15**

**21**

**28**

**36**

**45**

**55**

**fakultet=1 # første faktor kan ikke være null**

**for tall in range(1,6):**

**fakultet\*=tall # fakultet\*tall = neste fakultet**

**print(fakultet)**

**Resultat:**

**1 # 1\*1=1, 1!\*1=1!**

**2 # 1\*2=2, 1!\*2=2!**

**6 # 2\*3=6, 2!\*3=3!**

**24 # 6\*4=24, 3!\*4=4!**

**120 # 24\*5=120, 4!\*5=5! eller 5!=1\*2\*3\*4\*5=120**

**for tall in range(1,10):**

**kvadrat=tall\*\*2 # kvadrat=tall\*\*2**

**print(kvadrat)**

**Resultat:**

**1**

**4**

**9**

**16**

**25**

**36**

**49**

**64**

**81**

**for x in range(0,50): # gjelder fra og med 0 til og med 49**

**if x%2 == 1: # eks 5%2=1 fordi 5-2-2=1 i rest, 5 skal da skrives ut**

**print(x)**

**Resultat:**

**1**

**3**

**5**

**7**

**9**

**11**

**13**

**15**

**17**

**19**

**21**

**23**

**25**

**27**

**29**

**31**

**33**

**35**

**37**

**39**

**41**

**43**

**45**

**47**

**49**

**liste\_med\_tall=[5,1,7,3,8,2,1,6]**

**for tall in liste\_med\_tall:**

**print(tall\*\*2) # skriver ut kvadratallet til tallene I listen**

**Resultat:**

**25**

**1**

**49**

**9**

**64**

**4**

**1**

**36**

**for tall1 in range(1,11):**

**for tall2 in range(1,11): # tall1=1 og tall2=1,2,3,4,5,6,7,8,9,10**

**print(f"{tall1}\*{tall2}={tall1\*tall2}") # skriver først ut en gangen, deretter gjentas løkken med tall1=2 og tall2=1,2,3,4,5,6,7,8,9,10**

**Resultat:**

**1\*1=1**

**1\*2=2**

**1\*3=3**

**1\*4=4**

**1\*5=5**

**1\*6=6**

**1\*7=7**

**1\*8=8**

**1\*9=9**

**1\*10=10**

**.**

**.**

**.**

**10\*1=10**

**10\*2=20**

**10\*3=30**

**10\*4=40**

**10\*5=50**

**10\*6=60**

**10\*7=70**

**10\*8=80**

**10\*9=90**

**10\*10=100**

**def sporsmaal\_enkelt\_svar(sporsmaal, riktig\_svar):**

**svar = input(sporsmaal)**

**if svar == riktig\_svar:**

**print(f"Svaret {svar} er riktig!")**

**return 1**

**else:**

**print(f"Svaret {svar} er feil...")**

**return 0**

**# alternativer: ["Jehova", "Allah", "Buddha"]**

**# [["A", "Jehova"], ["B", "Allah"], ["C", "Buddha"]]**

**def sporsmaal\_multiple\_choice(sporsmaal, alternativer, riktig\_alternativ):**

**print(sporsmaal)**

**alternativ\_string = ""**

**print()**

**for indeks in range(len(alternativer)):**

**alternativ\_string += f"{indeks}: {alternativer[indeks]} "**

**svar = input(alternativ\_string)**

**# case 1: svar er en indeksr**

**# case 2: svar er en tekst**

**if svar == alternativer[riktig\_alternativ] or svar == str(riktig\_alternativ):**

**print(f"Svaret {svar} er riktig!")**

**return 1**

**else:**

**print(f"Svaret {svar} er feil...")**

**return 0**

**poengsum = 0**

**poengsum += sporsmaal\_multiple\_choice("Hva er hovedstaden i Frankrike?", ["Oslo", "Paris", "Berlin"], 1)**

**poengsum += sporsmaal\_enkelt\_svar("Hva er hovedstaden i Tyskland?", "Berlin")**

**poengsum += sporsmaal\_enkelt\_svar("Hva heter Gud i det gamle testamentet?", "Jehova")**

**print(f"Total poengsum er {poengsum}")**

**Resultat:**

**Hva er hovedstaden i Frankrike?**

**0: Oslo 1: Paris 2: Berlin 1**

**Svaret 1 er riktig!**

**Hva er hovedstaden i Tyskland?Berlin**

**Svaret Berlin er riktig!**

**Hva heter Gud i det gamle testamentet?Jehova**

**Svaret Jehova er riktig!**

**Total poengsum er 3**

**Oppgave: Sett på riktig fargekode. Forklar kodingen med # for den enkelte linje. Legg på flere spørsmål. Se om kodingen kan forbedres.**

**Turtle - peker**

**Import turtle # Skaper en peker som vi kan kontrollere.**

**turtle.color("blue")**

**turtle.forward(100)**

**turtle.left(90)**

**turtle.forward(100)**

**turtle.left(90)**

**turtle.forward(100)**

**turtle.left(90)**

**turtle.forward(100)**

**Resultet:**

**Tegner en blå kvadrat.**

**import turtle**

**turtle.circle(100)**

**turtle.penup()**

**turtle.forward(100)**

**turtle.pendown()**

**turtle.circle(100)**

**Resultat:**

**Tegner to sirkler som delvis ligger over hverandre.**

**import turtle**

**farge = input("Hvilken farge vil du ha?")**

**if farge == "grønn": # Alle if , elif og else hører sammen som en enhet.**

**farge = "green"**

**elif farge == "blå":**

**farge = "blue"**

**elif farge == "rød":**

**farge = "red"**

**else:**

**farge = "yellow"**

**turtle.color(farge)**

**rotasjon = 60**

**retning = input("Hvilken retning skal vi gå?")**

**if retning == "h":**

**turtle.right(rotasjon)**

**elif retning == "v":**

**turtle.left(rotasjon)**

**turtle.forward(100)**

**if retning == "h":**

**turtle.right(rotasjon)**

**elif retning == "v":**

**turtle.left(rotasjon)**

**turtle.forward(100)**

**Resultat:**

**Tegner en vinkel hvor du velger farge og retning.**

**Oppgave: Oppgrader programmet ved å legge inn en løkke og flere fargevalg.**

**import turtle**

**turtle.pensize(3)**

**turtle.color("blue")**

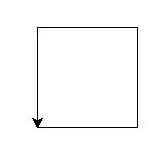
**for figur in range(4):**

**turtle.forward(100)**

**turtle.left(90)**

**Resultet:**

**Tegner et blått kvadrat.**

****

**Oppgave: Endre på betingelsene, som range, forward og left.**

**import turtle**

**for figur in range(4):**

**turtle.forward(25)**

**turtle.penup()**

**turtle.forward(50)**

**turtle.pendown()**

**turtle.forward(25)**

**turtle.left(90)**

**Resultat:**

**Tegner fire hjørner I et kvadrat.**

**Oppgave: Legg inn flere av den samme figuren utenpå hverandre. Kan man gjøre dette med en løkke? Legg også inn ulike farger på figurene.**

**Import turtle**

**star=turtle.Turtle() # Gir nytt navn på turtle.**

**for i in range(5):**

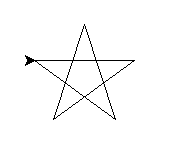
**star.forward(50)**

**star.right(144)**

**turtle.done()**

**Resultat:**

**Tegner en femkanta stjerne.**



**import turtle**

**spiral=turtle.Turtle()**

**for i in range(20):**

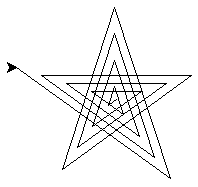
**spiral.forward(i\*10) # Første runde i løkka gir 0\*10=0, neste gir 1\*10=10 osv.**

**spiral.right(144)**

**turtle.done()**

**Resultat:**

**Tegner en femkanta spiralstjerne som gradvis øker i størrelse.**



**Import turtle**

**painter=turtle.Turtle()**

**painter.pencolor("blue")**

**for i in range(50):**

**painter.forward(50)**

**painter.left(123)**

**painter.pencolor("red")**

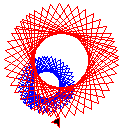
**for i in range(50):**

**painter.forward(100)**

**painter.left(123)**

**turtle.done()**

**Resultat:**



**Oppgave: Endre på betingelsen og variablene.**

**import turtle**

**polygon=turtle.Turtle() # Polygon er en mangekantet figure.**

**num\_sides=6 # Polygonet skal ha 6 kanter.**

**side\_length=70**

**angle=360.0/num\_sides # vinkel=360/6=60 grader**

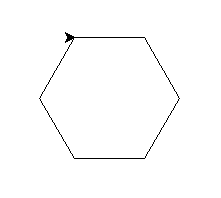
**for I in range(num\_sides): # range(6)**

**polygon.forward(side\_length)**

**polygon.right(angle)**

**turtle.done()**

**Resultat:**



**Oppgave: Lag et polynom som gradvis øker I størrelse.**

**from turtle import \***

**fillcolor(“purple”)**

**pensize(10)**

**pencolor(“black”)**

**forward(100)**

**begin\_fill()**

**forward(100)**

**left(90)**

**forward(100)**

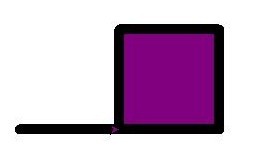
**left(90)**

**forward(100)**

**left(90)**

**end\_fill()**

**Resultat:**

****

**from turtle import \***

**penup() # Det skal ikke tegnes noen linjer.**

**for i in range(30,-1,-1): # Begynner med 30 går ned til 0.**

**stamp() # gir avtrykk av peker, turtle.**

**left(i) # Begynner med 30 grader deretter 29,28,27 osv for hver runde I løkka.**

**forward(20)**

**resultat:**

**Fungerer ikke??**

**import turtle**

**dot\_mønster=turtle.Turtle() # dot\_mønster er navnet på figuren**

**dot\_distance=25 # Skal du ha flere ord i en variable må du bruke underscore mellom ordene.**

**width=5**

**height=7 # Pennen må være oppe ellers vil den tegne en linje tilbake til utgangspunktet.**

**dot\_mønster.penup() # penup gjelder for begge løkkene.**

**for y in range(height): # Range(7), gir 7 rader.**

**for i in range(width): # range(5), gir 5 prikker per rad.**

**dot\_mønster.dot() # dot = setter en prikk**

**dot\_mønster.forward(dot\_distance) # Gir avstand mellom prikkene. Returnerer til andre løkke.**

**dot\_mønster.backward(dot\_distance\*width) # Første rad fullført. Går tilbake 25\*5=125**

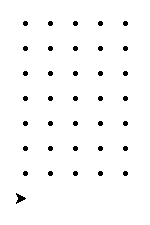
**dot\_mønster.right(90)**

**dot\_mønster.forward(dor\_distance)**

**dot\_mønster.left(90) # Klar for neste rad. Returnerer til første løkke.**

**turtle.done()**

**Resultat:**



**import turtle**

**hastighet = turtle.Turtle()**

**hastighet.speed(10)**

**for i in range(180): # Løkka gjentar det som står nedenfor 180 ganger.**

**hastighet.forward(100)**

**hastighet.right(30)**

**hastighet.forward(20)**

**hastighet.left(60)**

**hastighet.forward(50)**

**hastighet.right(30)**

**hastighet.penup()**

**hastighet.setposition(0,0) # Går tilbake til start.**

**hastighet.pendown()**

**hastighet.right(2) # 2\*180=360 grader**

**turtle.done()**

**«»»**

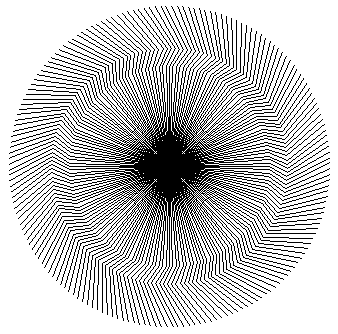
**turtle.setposition (x, y) vil sette skilpaddens posisjon til koordinatene du kobler til. (0, 0) ligger midt på skjermen - der skilpadden først startet. Legg merke til at du må sørge for at skilpadden er oppe, ellers vil den trekke en linje tilbake til start.**

**Du kan endre hastigheten på skilpadden ved å bruke turtle.speed. Hvis du setter hastigheten til 10, vil skilpadden gå veldig fort. Hvis du setter hastigheten til 1, vil skilpadden virkelig gå sakte (noe som er nyttig for å prøve å forstå hvor mye som gjøres). Hvis du setter hastigheten til null, vil skilpadden imidlertid gå til warpspeed og tegne så fort som den kan.**

**Hastigheten kan ikke være mindre enn 0 eller høyere enn 10.**

**«»»**

**Resultat:**



**Import turtle**

**colors=[“red”,”purple”,”blue”,”green”,”orange”,”yellow”] # Farge nr 0,1,2,3,4,5**

**turtle.speed(4)**

**turtle.bgcolor(“black”) # Gir sort bakgrunnsfarge.**

**for x in range(360):**

**turtle.pencolor(colors[x%6] # x%6 gir alle fargene, 0,1,2,3,4,5.**

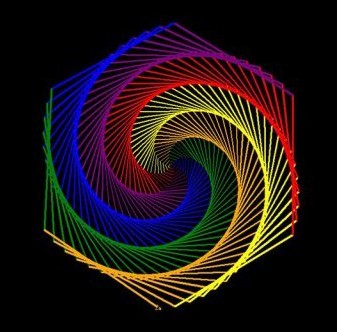
**turtle.width(x/100+1) # Streken blir bredere for hver runde I løkka.**

**turtle.forward(x) # Lengden på streken.**

**turtle.left(59) # Gir spiral effekt.**

**Resultat:**

**Økende størrelse på seksfarget polynom i spiral mønster.**

****

**x%6**

**0%6=0**

**1%6=1**

**2%6=2**

**3%6=3**

**4%6=4**

**5%6=5**

**6%6=0**

**7%6=1**

**8%6=2**

**.**

**.**

**.**

**Oppgave: Endre på variablene.**

**import turtle**

**turtle.speed(10)**

**for i in range(20):**

**turtle.circle(5\*i) # Gir økte sirkler for hver runde på den positive siden.**

**turtle.circle(-5\*i) # Gir økte sirkler for hver runde på den negative siden.**

**turtle.left(i) # Gir spiraleffekt.**

**Resultat:**

**Økende størrelse på sirkler i spiralmønster.**

**Oppgave: Endre på variablene og legg inn forskjellige farger.**

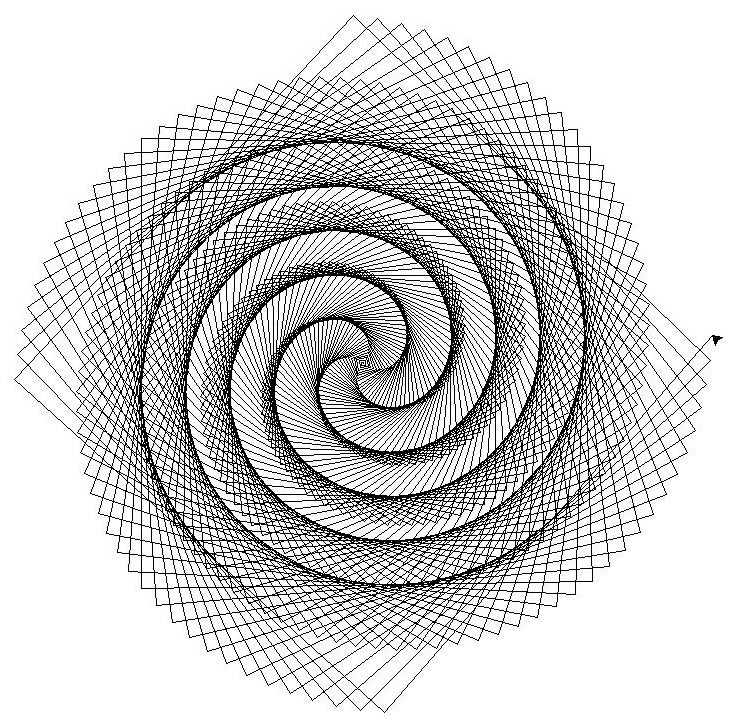
**from turtle import \* # Importerer alt fra turtle arkivet, alle kommandoer. Turtle modul.**

**for i in range(500): # Løkken gjentas 500 ganger.**

**forward(i) # Forward øker med en hver gang løkken gjentas.**

**left(91)**

**Resultat:**

****

**import turtle**

**turtle.Screen() # Gir et kvadratisk skjembilde.**

**turtle.speed(5)**

**turtle.bgcolor("blue") # Farger bakgrunnen blå.**

**turtle.title("Turtle") # Gir overskrift.**

**turtle.color("black") # Turtle tegner med svart farge.**

**def my\_sqrfunc(size): # my\_sqrfunc er navn på funksjonen.**

**for i in range(4):**

**turtle.fd(size) # fd=forward**

**turtle.left(90)**

**size = size – 5 # Reduserer lengden med 5 for hver løkke.**

**my\_sqrfunc(146) # Setter inn spesifikk verdi I funksjonen.**

**my\_sqrfunc(126)**

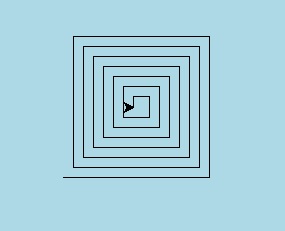
**my\_sqrfunc(106)**

**my\_sqrfunc(86)**

**my\_sqrfunc(66)**

**my\_sqrfunc(46)**

**my\_sqrfunc(26)**

**Resultat:**

**import turtle**

**turtle.speed(8)**

**turtle.color("blue")**

**side=400**

**turtle.penup()**

**turtle.goto(-200,-200)**

**turtle.pendown()**

**for i in range(1,100):**

**turtle.forward(side)**

**turtle.left(90)**

**side=side-4**

**Resultat:**

**Lager samme figur som over.**

**from turtle import\***

**from random import randint # Fra random modul importerer vi randint**

**speed(0)**

**bgcolor(“black”) # color, amerikansk stavemåte.**

**x=1 # Trenger en startverdi pga while løkka.**

**while x < 400: # Løkka gjelder så lenge variable x er under 400.**

**r=randint(0,225) # randint= random integer, velger tilfeldig heltall.**

**g=randint(0,225)**

**b=randint(0,225)**

**pencolor(r,g,b) # Gir fargekode til turtle.**

**fd(50+x) # fd=forward, legger på 1 for hver løkke.**

**rt(90.911) # Rotasjonen må være litt over 90 grader for å få spiral mønster.**

**x=x+1 # Øker med 1 for hver løkke til x=400.**

**Resultat:**

**Gir et fargerikt kvaratisk spiral mønster.**

**While løkke**

**tall = 0**

**while tall <5: # Løkken starter med 0 og går opp til og med 4. Betingelsen er mindre enn 5.**

**tall +=1 # tall + 1 = nytt tall, eks 0+1=1**

**print(tall) # Skriver ut 1. Løkken begynner på nytt ved å sjekke at 1 er mindre enn 5**

**Løkken går så lenge betingelsen gjelder, mindre enn 5.**

**Resultat:**

**1 # 0+1=1**

**2 # 1+1=2**

**3 # 2+1=3**

**4 # 3+1=4**

**5 # 4+1=5**

**liste\_med\_navn=["Ole","Per","Pål"] # indeks nr 0,1,2**

**indeks=0**

**while indeks < len(liste\_med\_navn): # Løkka forsetter så lenge indeks nr <3**

**print("Hei ",liste\_med\_navn[indeks]) # Setter inn navn etter indeks nr.**

**indeks +=1 # 0+1=1, returnerer 1 til while løkka, siden 1<3 vil løkka fortsette.**

**Resultat:**

**Hei Ole**

**Hei Per**

**Hei Pål**

**a=0**

**while a <21: # Betingelsen er tall<21, går gjennom alle tall fra 0 til 20.**

**if a%2 == 0: # tall som gir null i rest blir sendt videre til print.**

**print(a) # Alle tall som gir 0 i rest blir skrevet ut.**

**a+=1 # 0+1=1, gir neste tall som skal sjekkes i løkka.**

**Resultat:**

**0**

**2**

**4**

**6**

**8**

**10**

**12**

**14**

**16**

**18**

**20**

**penger=10000 # Start beløp.**

**rente=1.0345 # tilsvarer 3.45 %**

**år=0 # Starter med år null. Da du setter inn pengene i banken.**

**while penger<20000:**

**penger\*=rente # penger(år 0)\*rente=penger(år 1)**

**år +=1 # Penger etter år: 0+1=1**

**print(f"år {år} {penger:.2f} kr") # :.2f gir to desimaler.**

**print() # Gir tom mellomlinje hver gang løkken gjentas.**

**print(f"Det tar {år} år før du har over 20000 kr") # Beløp har oversteget betingelsen <20000, (år 21 20386.39 kr) siste år settes da inn i teksten. Gir tekst svar.**

**Resultat:**

**år 1 10345.00 kr**

**år 2 10701.90 kr**

**år 3 11071.12 kr**

**år 4 11453.07 kr**

**år 5 11848.20 kr**

**år 6 12256.97 kr**

**år 7 12679.83 kr**

**år 8 13117.29 kr**

**år 9 13569.83 kr**

**år 10 14037.99 kr**

**år 11 14522.30 kr**

**år 12 15023.32 kr**

**år 13 15541.63 kr**

**år 14 16077.81 kr**

**år 15 16632.50 kr**

**år 16 17206.32 kr**

**år 17 17799.93 kr**

**år 18 18414.03 kr**

**år 19 19049.32 kr**

**år 20 19706.52 kr**

**år 21 20386.39 kr**

**Det tar 21 år før du har over 20000 kr**

**Oppgave: Endre på betingelsen og variablene.**

**fasit1 ="Lisboa"**

**fasit2 ="lisboa"**

**svar = input("Hva er hovedstaden i Portugal? ")**

**while svar != fasit1 and svar != fasit2: # Løkka gjelder så lenge du ikke gir riktig svar.**

**svar = input("Feil svar! Prøv igjen. ") # != betyr at ditt svar er ikke lik fasit.**

**print("Helt riktig!") # Skrives ut hvis du har svart riktig. Løkken brytes.**

**Resultat:**

**Hva er hovedstaden i Portugal? berlin**

**Feil svar! Prøv igjen. Lisboa**

**Helt riktig!**

**Sannsynlighet**

**from random import \* # Importerer tilfeldig trekking av tall og alt om sannsynlighetsregning.**

**antall\_enere=0**

**N=50**

**for i in range(N): # Løkka gjelder for range(0,50), gir 50 tilfeldige tekninger.**

**tall=randint(1,7) # randint betyr at det trekkes tilfeldig tall: 1,2,3,4,5,6,7**

**print(tall)**

**if tall==1: # Teller opp alle tilfeldige tall som er 1.**

**antall\_enere +=1 # Begynner med 0+1=1.**

**andel\_enere=100\*antall\_enere/N # Regner ut andel i prosent.**

**print(f"Etter {N} kast har vi fått {antall\_enere}, det vil si {andel\_enere}%.") #Gir tekstsvar etter 50 tilfeldige trekninger**

**Resultat:**

**7**

**3**

**5**

**6**

**1**

**7**

**1**

**.**

**.**

**.**

**7**

**7**

**4**

**3**

**1**

**3**

**Etter 50 kast har vi fått 7, det vil si 14.0%.**

**Plot**

**from matplotlib.pyplot import \* # Importerer alt for dataoppsett.**

**xverdier=[]**

**yverdier=[]**

**a=2**

**b=3**

**for i in range(50):**

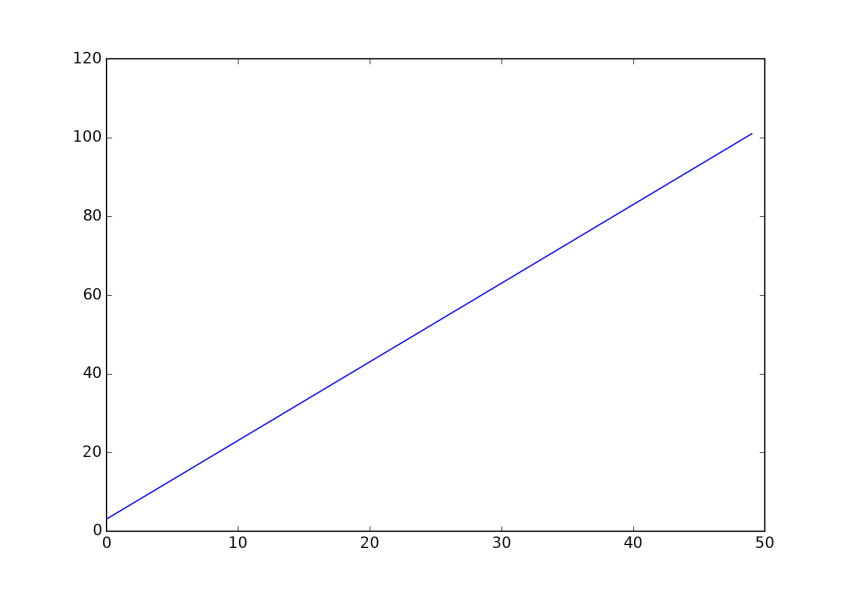
**xverdier.append(i)**

**yverdier.append(a\*i+b)**

**plot(xverdier,yverdier)**

**show()**

**Resultat:**

****

**from matplotlib.pyplot import \* # Importerer alt for dataoppsett, markert med \***

**xverdier=[] # Tom liste som kan fylles opp med bruk av append.**

**yverdier=[]**

**for i in range(50):**

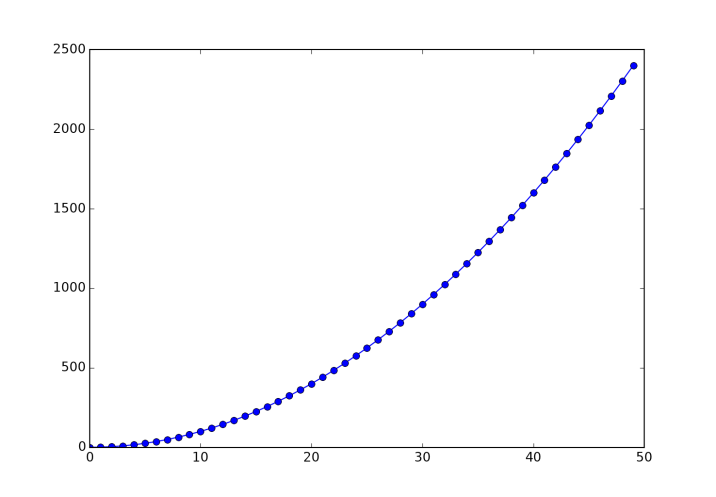
**xverdier.append(i) # Gir x verdier som settes i lista.**

**yverdier.append(i\*\*2) # Regner ut y (kvadrattall) verdien som settes i listen.**

**plot(xverdier,yverdier,"o-") # Gir koordinatene + at de skal markes som kuler.**

**show() # Viser grafen.**

**Resultat:**

****

**from matplotlib.pyplot import \***

**xverdier\_ruter=[]**

**yverdier\_riskorn=[]**

**yverdier\_kilo=[]**

**for i in range(1,65):**

**xverdier\_ruter.append(i)**

**yverdier\_riskorn.append(2\*\*(i-1))**

**yverdier\_kilo.append(2\*\*(i-1)/50000)**

**plot(xverdier\_ruter,yverdier\_riskorn, "o-",label="riskorn")**

**plot(xverdier\_ruter,yverdier\_kilo, "o-",label="kilo")**

**xlabel("X-riskorn")**

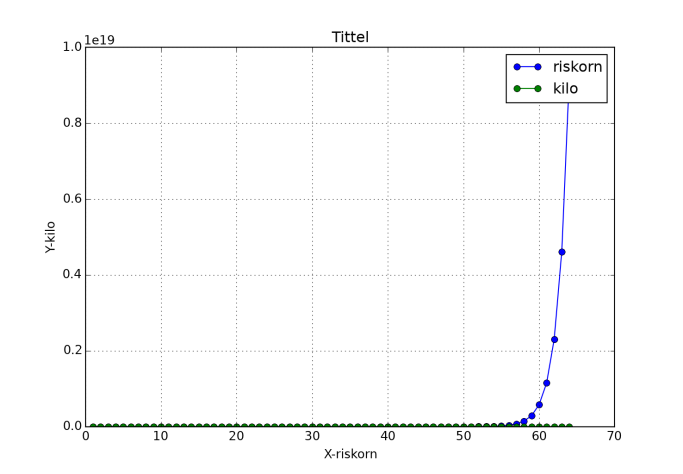
**ylabel("Y-kilo")**

**title("Tittel")**

**legend() # Boks som viser navn og fargekode på den enkelte funksjon.**

**grid() # Gir rutemønster.**

**show() # Viser grafen.**

**Resultat:**

**from matplotlib.pyplot import \* # Importerer alt til python plot/graf.**

**xverdier=[]**

**yverdier\_lineær=[]**

**yverdier\_kvadratisk=[]**

**yverdier\_kubisk=[]**

**for i in range(10): # Gir 10 koordinatpunkter.**

**xverdier.append(i)**

**yverdier\_lineær.append(2\*i) # Gir 2i**

**yverdier\_kvadratisk.append(2\*i\*\*2)**

**yverdier\_kubisk.append(i\*i\*i)**

**plot(xverdier,yverdier\_lineær,"o-",label="Lineær") # label gir navnelapp.**

**plot(xverdier,yverdier\_kvadratisk, "o-",label="Kvadratisk")**

**plot(xverdier,yverdier\_kubisk, "o-",label=r"$x^2$") # $x^2$ gir x2 i label.**

**xlabel("X-verdier") # Gir navn på x-aksen**

**ylabel("Y-verdier")**

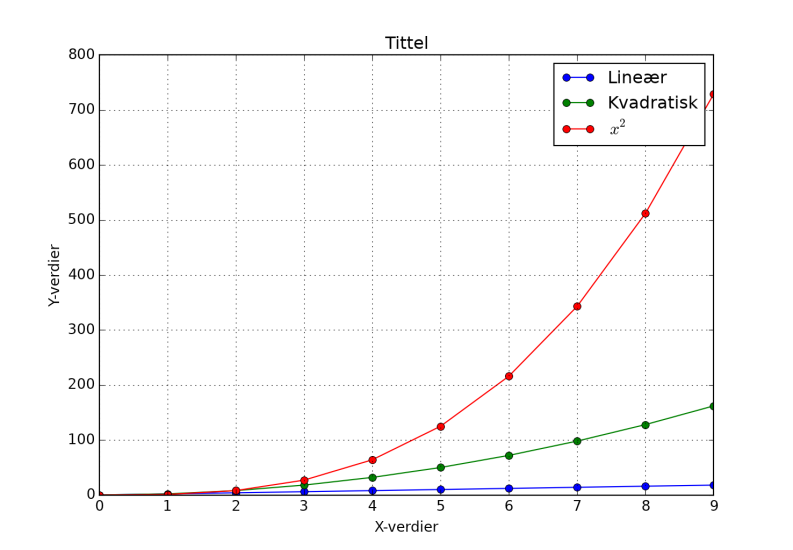
**title("Tittel") # Gir tittel på graf.**

**legend() # Boks som viser navn og fargekode på den enkelte funksjon.**

**grid() # Gir rutemønster.**

**show()**

**Resultat:**

****

**import matplotlib.pyplot as plt # Definerer python plot/graf som plt.**

**x = [1,2,3,4,5,6] # x akse verdier.**

**y = [2,4,1,5,2,6] # y akse verdier.**

**plt.plot(x, y, color='green', linestyle='dashed', linewidth = 3, marker='o', markerfacecolor='blue', markersize=12) # 'o' gir kuleformet koordinatpunkter. Det er også bestemt at de skal være blå og ha størrelse 12.**

**plt.ylim(1,8) # Begrenser y aksen.**

**plt.xlim(1,8) # Begrenser x aksen.**

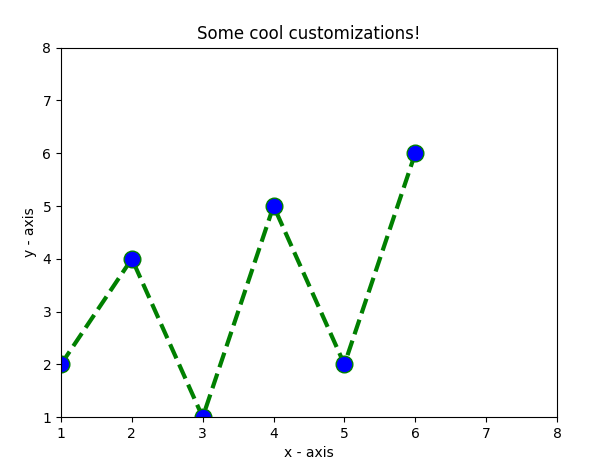
**plt.xlabel('x - axis') # Gir navn på x aksen.**

**plt.ylabel('y - axis') # Gir navn på y aksen.**

**plt.title('Some cool customizations!') # Tittel på grafen.**

**plt.show() # Viser grafen.**

**Resultat:**

****

**import matplotlib.pyplot as plt**

**x= [1, 2, 3, 4, 5] # x koordinat til søylene.**

**height = [10, 24, 36, 40, 5] # Høyden på søylene.**

**plt.bar(x, height, width = 0.8, color = ['red', 'green']) #Lage søylene. Width er bredden på søylene. Color gir farge på søylene.**

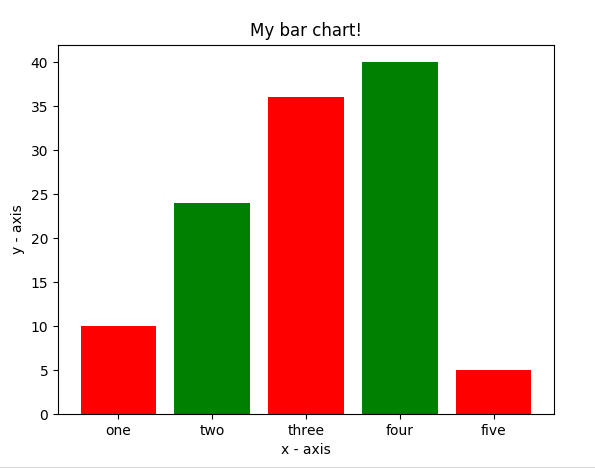
**plt.xlabel('x - axis') # Navn på x aksene.**

**plt.ylabel('y - axis') # Navn på y aksen.**

**plt.title('My bar chart!') # Tittel på grafen.**

**plt.show()**

**Resultat:**

****

**Funksjoner**

**def kvadrat(x): # Lager en ny funksjon hvor navnet er kvadrat.**

**return x\*\*2 # Forklarer hva funksjonen skal gjøre og at resultatet skal returneres.**

**print(kvadrat(5)) # Du bruker funksjonen din og har bestemt at den ukjente er lik 5.**

**Resultat:**

**25**

**def kubikk(x):**

**return x\*\*3**

**print(kubikk(5))**

**Resultat:**

**125**

**def hei(navn):**

**a="Hei "**

**b=navn**

**return a+b**

**print(hei("Espen"))**

**Resultat:**

**Hei Espen**

**def finn\_trekant\_vinkel(vinkel1,vinkel2):**

**vinkel3=180-vinkel1-vinkel2**

**return vinkel3**

**print(finn\_trekant\_vinkel(30,60))**

**Resultat:**

**90**

**def minus(a,b):**

**sum=a-b**

**return sum**

**print(minus(3,2))**

**Resultat:**

**1**

**def pluss(a,b):**

**sum=a+b**

**return sum**

**print(pluss(2,3))**

**Resultat:**

**5**

**def oddetall(tall):**

**regnestykke=2\*tall+1**

**return (regnestykke)**

**print(oddetall(5))**

**Resultat:**

**11**

**def main():**

**def oddetall(tall):**

**return 2\*tall+1**

**print(oddetall(5))**

**print(oddetall(10))**

**main() # main definerer en funksjon med spesifike verdier.**

**Resultat:**

**11**

**21**

**Legge inn kommentar**

**# Dette er en kommentar. Gjelder for resten av linja. Blir ikke en del av koden.**

**”””**

**Dette er en kommentar over flere linjer.**

**Dette er en kommentar over flere linjer.**

**Dette er en kommentar over flere linjer.**

**”””**

**Spill**

**Playground, Learn to code 1 and 2. Bra spill for nybegynnere. Lærer kodespråket. Passer for barn og ungdom.**

**Code combat. Bra spill for nybegynnere av Pyton.**

**Undervisning**

**Pycharm. Kodeskole for Pyton. Intelligent program som hjelper deg med å kode og finne feil.**

**Python tutorial. Bra forklart med eksempler.**

**NTNU pyton grunnkurs TDT4110 IT. Bra forklart med eksempler.**

**Kommandoer**

**Type data:**

**String # "dette er en streng" tekstverdi**

**Integer(int) # 13, heltall**

**Boolean(bool) # True eller False**

**Float # 0.435, desimaltall**

**None # None=ingenting**

**= # brukes til å gi en variabel verdi**

**== # en verdi er lik en annen**

**!= # en verdi er IKKE lik en annen**

**< # er mindre enn**

**> # er større enn**

**<= # er mindre eller lik**

**> # er større eller lik**

**% # modulo - gir resten av en divisjon**

**/ # divisjon**

**+ # addisjon**

**- # substraksjon**

**\* # multiplikasjon**

**\*\* # opphøyer et tall**

**is # sjekker om to objekter er like**

**in # sjekker om en verdi er i en liste**

**not # inverterer is og in**

**None # en verdi som er ingenting**

**for # løkke med bestemt lengde**

**while # løkke som går så lenge betingelsen gjelder**

**def # lager din egen funksjon**

**print # printer til terminalen**

**variabel, # holder på en verdi**

**len # lengde, antall**

**boolean, # sant / usant - verdi**

**input, # printer til terminalen, og tar inn fra tastaturet helt til enter trykkes**

**if # sjekker betingelse, kjøres kun om uttrykket er sant**

**elif # kjøres etter forrige if/elif**

**else # hvis ingen andre var sanne**

**liste # liste som [1, 5, 3, 10]**

**i # index**

**pass # brukes når en operasjon eller del av programmet ikke skal tas med, står over**

**print() # lager tom mellomlinje**

**and # Begge er true**

**or # En eller begge er true**

**and**

**true false**

**true 1 0**

**false 0 0**

**or**

**true false**

**true 1 1**

**false 1 0**

**Turtle kommandoer**

**Turtle() # Oppretter og returnerer et nytt skilpadde objekt.**

**forward() # Flytter skilpadden frem med det angitte tallet.**

**backward() # Flytter skilpadden bakover med det angitte tallet.**

**right() # Vender skilpadden med klokken.**

**left() # Vender skilpadden mot klokken.**

**penup() # Henter skilpaddens penn.**

**pendown() # Setter skilpaddens penn ned.**

**up() # Henter skilpaddens penn.**

**down() # Setter skilpaddens penn ned.**

**color() # Endrer fargen på skilpaddens penn.**

**fillcolor() # Endrer fargen som skilpadden vil bruke til å fylle et polygon.**

**heading() # Returnerer gjeldende overskrift.**

**position() # Returnerer gjeldende posisjon.**

**goto(x,y) # Flytter skilpadden til posisjon (x, y).**

**begin\_fill() # Husker startpunktet for et fullført polygon.**

**end\_fill() # Polygonet er fullført og fylles med følgende farge.**

**dot() # Lager dot ved gjeldende posisjon.**

**stamp() # Etterlater et skilpaddeformet avtrykk på det nåværende stedet.**

**shape() # Form på pekeren, kan være ‘arrow’, ‘classic’, ‘turtle’ or ‘circle’**

**bgcolor() # Gir bakgrunnsfarge.**

**Width() # Gir bredde på streken.**

**goto() # Flytter skilpadden til de gitte x- og y-koordinatene. Hvis pennen er nede vil det bli trukket en linje fra de forrige koordinatene til de nye koordinatene.**

**Undo() # Fjerner siste steget. Kan brukes mange ganger for eksempel i en løkke.**

**fillcolor() # Angir fargen på den utfylte formen når end\_fill () kalles.**