Print() Resultaat weergeven (pint) Print("Hello, world!") Int() geeft de parameter waarde terug als integer Int(7.5) Float() geeft de parameter waarde terug als kommagetal Float(7) Str() geeft de parameter waarde terug als string vaarde van code vaarde vaarde vaarde weer vaarde weer vaarde weer vaarde weer vaarde weer vaarde weer vaarde de waarde van een vaarde van	Instructie	Betekenis	Voorbeeld
Int() → Type Casting Float() → Type Casting Float() → Type Casting Str() → Type Casting Syntax Errors Syntax Errors Fouten in code zodat deze niet uitgevoerd kan worden An worden Fouten tijdens uitvoeren van code Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde Foutuneert de grootste parameter Waarde van een waarde Max() Retourneert de kleinste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Fouten tijdens uitvoeren van code Abs(-5) Abs(-5) Max(2, 10, 5) Geeft de lengte van een waarde een parameter Round(getal, decimalen) Fouten tijdens uitvoeren van code Abs(-5) Waarde van een waarde Waarde van een waarde Waarde Min(2, 10, 5) Befinieer een een eindwaarde pirint Print(, sep = <string>) Definieer een eindwaarde bij de print F'(a:uittijnijng ruimte Print(":>5.2f''.format(3.784))</string>	Print()	Resultaat weergeven	Print("Hello, world!")
## Type Casting warde terug als integer Float() geeft de parameter waarde terug als kommagetal Str() geeft de parameter yaarde terug als kommagetal Str() geeft de parameter yaarde terug als string Syntax Errors Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft net dataype van een waarde weer Pow(basis, exponent, modulo) Verheft de basis tot de exponent en neemt er eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een eindwaarde bij de print F'(a:uitlijnijng ruimte) Geeft een variabele a Print("{:>5.2f}".format(3.784))</string>	· · · · · · · · · · · · · · · · · · ·	_	
Float() → Type Casting geeft de parameter waarde terug als kommagetal Str() → Type Casting geeft de parameter waarde terug als string Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten in jdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow() basis, exponent, modulo) Verheft de basis tot de exponent en neemt er eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een eindwaarde bij de print F'(a:uitlijnijng ruimte Geeft de en variabele a Float(7) Str(8) Frint("Hello, world! Str(8) Frint("Hello, world! Str(8) Str(8) Str(8) Str(8) Str(8) Str(8) Str(8) Str(9) Str(9) Str(9) Str(9) Str(9) Str(9)</string>	**	I = -	,
Float() → Type Casting Str() geeft de parameter waarde terug als kommagetal Str() geeft de parameter waarde terug als string Syntax Errors Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Verheft de basis tot de exponent en neemt er eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de grootste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraggt de gebruiker naar input Print(, sep = <string>) Definieer een eindwaarde bij de print F'(a:uitlijnijng ruimte Geeft een variabele a Fouten tirogads tring Str(8) Str(8) Str(8) Str(8) Print("Hello, world!) Str(8) Print("Hello, world!) Str(8) Frint("Hello, world!) Frint("Hello, world!) Str(8) Frint("Hello, world!) Str(8) Frint("Hello, world!) Frint("Hello, world!) Frint("Hello, world!) Frint("Hello, world!) Str(8) Frint("Hello, world!) Frint("Hello, world!) Frint("Hello, world!) Frint("Hello, world!) Frint("Hello, world!) Frint("Hello, world! Frint("Hello, w</string>	,, ,	_	
Str() Str() geeft de parameter waarde terug als string Syntax Errors Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Verheft de basis tot de exponent en neemt er eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een gen parameter terug Input() Vragt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print F''(a:uitlijinjing ruimte Sert terug als string Str(8) Str(8) Str(8) Str(8) Str(8) Str(8) Str(8) Str(8) Str(8) Print("Hello, world!) Print("Hello, world! Print("Hello, world!) Print("Hello, world! Ascondanted Ascondanted</string>	Float()	geeft de parameter	Float(7)
Str() → Type Casting Syntax Errors Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde Fetourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Round(getal, decimalen) Frint(, sep = <string>) Definieer een separator bij de print F''(a:uitlijnjing ruimte Fouten tin code zodat deze niet out oode zodat deze niet uitgevoerd kan world!) Print(, sen = string>) Fouten tin code zodat deze niet oode zodat deze niet uitgevoerd kan world!) Print(, sen = = "") Frint(, end = <string>) Fouten tin code zodat deze niet oode zodat deze niet oode zodat deze niet uitgevoerd kan world!) Print(, sen = = "") Frint(, end = = "") Fouten tin code zodat deze niet uitgevoerd kan world!) Print("Hello, world! Print("Hello Print</string></string>	→ Type Casting	waarde terug als	
Syntax Errors Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde van een waarde waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Print(, sep = <string>) Definieer een eindwaarde bij de print F''{a:uitlijnijng ruimte} Print(":>5.2f}".format(3.784))</string>		kommagetal	
Fouten in code zodat deze niet uitgevoerd kan worden Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde van waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een geen parameter terug Input() Vraagt de gebruiker naar input Print(, end = <string>) Definieer een eindwaarde bij de print F'(a:uitlijinjing ruimte) Fouten in code zodat dezenet uitgevoerd kan wordel!) Print("Hello, world!) Print("Hello, world!) Print("Hello, world!) Print("Hello, world!) Print("Hello, world!) Prove den van code Type(a) Prove(2, 3, 3) Pow(2, 3, 3) Pow(2, 3, 3) Abs(-5) Max(2, 10, 5) Max(2, 10, 5) Max(2, 10, 5) Round(3.1356543, 3) Op een aantal parameter Round(3.1356543, 3) Op een aantal parameters Len("Patrick") Print(, sep = "x") Print(, sep = "x") Print(, sep = "x") Print(, end = <"")</string>	Str()	geeft de parameter	Str(8)
Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde van een waarde Fouten de grootste parameter Min() Retourneert de grootste parameter Round(getal, decimalen) Rond een waarde van een waarde sop een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print F"4a:uitlijnijng ruimte ZeroDivisionError ZeroDivisionEror</string>	Type Casting	waarde terug als string	
Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde Max() Min() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een eindwaarde bij de print F"(a:uitlijnjing ruimte) Fint(, end = <string>) Geeft de nataye van zeparator bij de print Fire (a:uitlijnjing ruimte) ZeroDivisionError Type(a) ZeroDivisionError Type(a) Abs(-5) Pow(2 ,3, 3) Pow(2 ,3, 3) Abs(-5) Max(2, 10, 5) Min(2, 10, 5) Min(2, 10, 5) Min(2, 10, 5) Min(2, 10, 5) Len("Patrick") Een ("Patrick") Print(, sep = "x") Separator bij de print Print(, end = = "") eindwaarde bij de print Print("{:>52.2f}".format(3.784))</string></string>	Syntax Errors	Fouten in code zodat	Print("Hello, world!)
Runtime Errors Fouten tijdens uitvoeren van code Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde van een waarde van een waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separato bij de print F"{a:uitlijnijng ruimte} Jrype(a) Type(a) Fow(2, 3, 3) Abs(-5) Max(2, 10, 5) Min(2, 10, 5) Min(2, 10, 5) Round(3.1356543, 3) Open aantal parameters En("Patrick") Print(, sep = "X") Separator bij de print Print(, end = "") Frint(, end = "") Frint(, end = "")</string>		deze niet uitgevoerd	
Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een waen en parameter terug Input() Print(, sep = <string>) Definieer een eindwaarde bij de print F"{a:uitlijnijng ruimte} Type(a) Envice, 3, 3) Abs(-5) Max(2, 10, 5) Min(2, 10, 5) Round(3.1356543, 3) Op een aantal parameter Round(3.1356543, 3) Input("Geef een getal: ") Print(, sep = "x") Separator bij de print Print(, end = = "") Frint(, end = = "")</string>		kan worden	
Type() Geeft het dataype van een waarde weer Pow(basis, exponent, modulo) Abs() Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print F''{a:uitlijnijng ruimte} Type(a) Type(a) Pow(2 ,3, 3) Pow(2 ,3, 3) Pow(2 ,3, 3) Abs(-5) Max(2, 10, 5) Min(2, 10, 5) Min(2, 10, 5) Len("Patrick") Pount("Fatrick") Print(, sep = "x") Separator bij de print Print(, end = <"") Print(, end = = "") Frint(, end = = "")</string></string>	Runtime Errors	Fouten tijdens	ZeroDivisionError
Pow(basis, exponent, modulo) Pow(basis, exponent, modulo) Verheft de basis tot de exponent en neemt er eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print F"{a:uitlijnijng ruimte} een waarde weer Pow(2 ,3, 3) Pow(2 ,3, 3) Pow(2 ,3, 3) Pow(2 ,3, 3) Abs(-5) Abs(-5) Max(2, 10, 5) Min(2, 10, 5) Len("Patrick") Len("Patrick") Print(, sep = separator bij de print Print(, sep = "x") Print(, end = = "") Print(, end = = "") Frint(, end = = "")</string></string>		uitvoeren van code	
Pow(basis, exponent, modulo) Verheft de basis tot de exponent en neemt er eventueel de modulo van	Type()	Geeft het dataype van	Type(a)
exponent en neemt er eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Print(, sep = <string>) Definieer een eindwaarde bij de print F"{a:uitlijnijng ruimte} Exponent en neemt er een waarde af op een variabele a print ("{s>5.2f}".format(3.784))</string>		een waarde weer	
eventueel de modulo van Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte} Abs(-5) Abs(-5 Abs(-5) Abs(-5) Abs(-5 Abs(-5) Abs(-5 Abs(-5) Abs(-5 Abs(-5) Abs(-5 Abs(-5) Abs(-5 Abs(-1, -1, -1, -1, -1, -1, -1,</string></string>	Pow(basis, exponent, modulo)	Verheft de basis tot de	Pow(2,3,3)
Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte} Vaage van een waarde af Abs(-5) Max(2, 10, 5) Min(2, 10, 5) Min(2, 10, 5) Len("Patrick") Round(3.1356543, 3) Len("Patrick") Enput("Geef een getal: ") Print(, sep = "x") Print(, sep = "x") Print(, end = = "") Frint(, end = = "")</string>		exponent en neemt er	
Abs() Geeft de absolute waarde van een waarde Max() Retourneert de grootste parameter Min() Retourneert de kleinste parameter Round(getal, decimalen) Round(getal, d		eventueel de modulo	
WaardewaardeMax()Retourneert de grootste parameterMax(2, 10, 5)Min()Retourneert de kleinste parameterMin(2, 10, 5)Round(getal, decimalen)Rond een waarde af op een aantal parametersRound(3.1356543, 3)LenGeeft de lengte van een parameter terugLen("Patrick")Input()Vraagt de gebruiker naar inputInput("Geef een getal: ")Print(, sep = <string>)Definieer een separator bij de printPrint(, sep = "x")Print(, end = <string>)Definieer een eindwaarde bij de printPrint(, end== "")f"{a:uitlijnijng ruimteGeeft een variabele aPrint("{:>5.2f}".format(3.784))</string></string>		van	
Max()Retourneert de grootste parameterMax(2, 10, 5)Min()Retourneert de kleinste parameterMin(2, 10, 5)Round(getal, decimalen)Rond een waarde af op een aantal parametersRound(3.1356543, 3)LenGeeft de lengte van een parameter terugLen("Patrick")Input()Vraagt de gebruiker naar inputInput("Geef een getal: ")Print(, sep = <string>)Definieer een separator bij de printPrint(, sep = "x")Print(, end = <string>)Definieer een eindwaarde bij de printPrint(, end = = "")f"{a:uitlijnijng ruimteGeeft een variabele aPrint("{:>5.2f}".format(3.784))</string></string>	Abs()	Geeft de absolute	Abs(-5)
Max()Retourneert de grootste parameterMax(2, 10, 5)Min()Retourneert de kleinste parameterMin(2, 10, 5)Round(getal, decimalen)Rond een waarde af op een aantal parametersRound(3.1356543, 3)LenGeeft de lengte van een parameter terugLen("Patrick")Input()Vraagt de gebruiker naar inputInput("Geef een getal: ")Print(, sep = <string>)Definieer een separator bij de printPrint(, sep = "x")Print(, end = <string>)Definieer een eindwaarde bij de printPrint(, end = = "")f"{a:uitlijnijng ruimteGeeft een variabele aPrint("{:>5.2f}".format(3.784))</string></string>		waarde van een	
Min()Retourneert de kleinste parameterMin(2, 10, 5)Round(getal, decimalen)Rond een waarde af op een aantal parametersRound(3.1356543, 3)LenGeeft de lengte van een parameter terugLen("Patrick")Input()Vraagt de gebruiker naar inputInput("Geef een getal: ")Print(, sep = <string>)Definieer een separator bij de printPrint(, sep = "x")Print(, end = <string>)Definieer een eindwaarde bij de printPrint(, end= = "")f"{a:uitlijnijng ruimteGeeft een variabele aPrint("{:>5.2f}".format(3.784))</string></string>		waarde	
Min() Retourneert de kleinste parameter Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print F''{a:uitlijnijng ruimte Min(2, 10, 5) Round(3.1356543, 3) Popen aental parameter Len("Patrick") Input("Geef een getal: ") Print(, sep = "x") Print(, end = = "") Print(, end = = "") Print(, end = = "")</string></string>	Max()	Retourneert de	Max(2, 10, 5)
Round(getal, decimalen) Rond een waarde af op een aantal parameters		grootste parameter	
Round(getal, decimalen) Rond een waarde af op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een eindwaarde bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Round(3.1356543, 3) Round(3.1356543, 3) Len("Patrick") Input("Geef een getal: ") Print(, sep = "x") Print(, sep = "x") Print(, end = = "") Print(, end = = "")</string></string>	Min()	Retourneert de	Min(2, 10, 5)
op een aantal parameters Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Geeft een variabele a Print("{:>5.2f}".format(3.784))</string></string>			
Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Geeft een variabele a Print("{:>5.2f}".format(3.784))</string></string>	Round(getal, decimalen)	Rond een waarde af	Round(3.1356543, 3)
Len Geeft de lengte van een parameter terug Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Geeft de lengte van Len("Patrick") Input("Geef een getal: ") Print(, sep = "x") Print(, sep = "x") Print(, end = = "") Print(, end = = "")</string></string>		op een aantal	
Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte een parameter terug Input("Geef een getal: ") Print(, sep = "x") Print(, sep = "x") Print(, end = = "") Print(, end = = "") Print("{:>5.2f}".format(3.784))</string></string>		•	
Input() Vraagt de gebruiker naar input Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print Print(, end = <string>) Print(, end = = "") f"{a:uitlijnijng ruimte Vraagt de gebruiker Input("Geef een getal: ") Print(, sep = "x") Print(, end = = "") Print(, end = = "") Frint("{:>5.2f}".format(3.784))</string></string></string>	Len	· ·	Len("Patrick")
Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Print(, end = = "") Refinieer een eindwaarde bij de print F"{a:uitlijnijng ruimte Print("{:>5.2f}".format(3.784))</string></string>			
Print(, sep = <string>) Definieer een separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Definieer een eindwaarde bij de print Geeft een variabele a Print(, sep = "x") Print(, end = = "") Print(, end = = "")</string></string>	Input()		Input("Geef een getal: ")
separator bij de print Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Seeft een variabele a Print("{:>5.2f}".format(3.784))</string>		'	
Print(, end = <string>) Definieer een eindwaarde bij de print f"{a:uitlijnijng ruimte Definieer een eindwaarde bij de print Frint(, end = = "") Print(, end = = "") Print(, end = = "")</string>	Print(, sep = <string>)</string>		Print(, sep = "x")
eindwaarde bij de print f"{a:uitlijnijng ruimte Geeft een variabele a Print("{:>5.2f}".format(3.784))			
f"{a:uitlijnijng ruimte print Geeft een variabele a Print("{:>5.2f}".format(3.784))	Print(, end = <string>)</string>		Print(, end= = "")
f"{a:uitlijnijng ruimte Geeft een variabele a Print("{:>5.2f}".format(3.784))		•	
		•	
.decimalen variabeletype}" terug in combinatie			Print("{:>5.2f}".format(3.784))
,, ,	.decimalen variabeletype}"		
van aanlijning, precisie			
f'(string1" if a <conditie> else en decimalen en f'Is $\{x\}$ even? $\{"Ja" if x\%2 == 0 else \}$</conditie>			1
"string2")' eventueel een conditie "Nee"}"	"string2")'	eventueel een conditie	"Nee"}"
Import/ From import Importeer (een deel Import math	Import/ From import	Importeer (een deel	Import math
van) een module	porc, rrom miporc	· ·	- mpore macin
Math.exp() Retourneert 'e' tot Math.exp(3)	Math.exp()		Math.exp(3)
parameter			

Math.log()	Retourneert natuurlijk logaritme van parameter	Math.log(12)
Math.log10()	Retourneert 10- logartime van parameter	Math.log10(1000)
Math.sqrt()	Retourneert vierkantswortel van parameter	Math.sqrt(144)
Random.random()	Retourneert random float uit [0,1[Radnom.random()
Random.randint(x, y)	Retourneert parameter uit [x,y]	Random.randint(2, 5)
Random.seed()	Inititialiseert de toevalsgenerator	Random.seed(1)
Sys.exit()	Sluit het hele programma af met een 'error'	Sys.exit()
Range(startgetal, eindgetal, stapgrootte)	Iteratie over [startgetal, eindgetal[met gegeven stapgrootte	Range(1, 10, 2)
Continue()	Herstart huidige loop	Continue()
Break()	Breekt uit huidige loop	Break()
Def naamfunctie(): <actie></actie>	Initialiseert een functie	Def bereken_som():
Return()	Zorgt ervoor dat de functie een waarde teruggeeft	Return(som_kwadraten)
Isinstance(parameter, datatype)	Checkt of een parameter een bepaald datatype is	Isinstance(5, int)
Global	Zorgt ervoor dat globale variabele kan gewijzigd worden binnen functie	Global x
lambda <parameters>: <actie></actie></parameters>	Initialiseert een anonieme functie bestaande uit 1 regel code	f = lambda x: x*x
Pass	Plaatsvuller die niets doet	Pass
type() /class	Retourneert informatie over de klasse	pclass
init()	Zorgt dat de functie voor iedere instantie van de klasse wordt toegepast	definit(self):
str() /repr()	Formatteert output van klasse	defrepr(self):

eq(self,other)	Zorgt ervoor dat ==	defeq(self,other):
,	operator gebruikt kan	
	worden	
Assert, 'message'	Checkt of de	assert breedte > 0 and hoogte > 0,
, 3	voorwaarde voldoet	'ongeldige rechthoek'
	en geeft anders een	
	error message	
Copy.copy()	Maakt een hard copy	Copy.copy(punt)
	van een parameter	
\n	Teken voor nieuwe	\n
	regel	`
\t	Teken voor inspringing	\p
String[a:b:c]	Print substring af die	"josefien"[2:6:2]
of a state of	begint bij a, eindigt bij	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	b-1 en stapgrootte c	
String.strip()	Verwijdert alle spaties,	" En nu iets heel anders \n ".strip()
	newline tekens, aan	
	begin en einde van	
	string	
String.find("a",b,c)	Zoekt in de string naar	"barabara".find("bar")
	eerste voorkomen van	·
	string a vanaf b tot c,	
	anders -1	
String.replace("a","b",c)	Vervangt in de string c	"barbara".replace("bar","ber",2)
	keer a in b	
String.split("a", aantal splits)	Maakt een lijst van de	"Jos eet kaas".split(" ")
Re.split('splitter', string)	delen van de string ma	
	a als separator	
String.isalpha()	Retourneert true als	"Jef".isalpha()
	alle tekens letters zijn	
String.isdigit()	Retourneert true als	"47484".isdigit()
	alle tekens cijfers zijn	
String.isspace()	Retourneert true als	" \n".isspace()
	alle tekens spaties of	
	newlinetekens zijn	
String.rstrip()	Verwijdert alle spaties,	"En nu iets heel anders \n ".rstrip()
	newline tekens, aan	
	einde van string	
String.lstrip()	Verwijdert alle spaties,	" En nu iets heel anders".lstrip()
	newline tekens, aan	
	begin van string	
String.rfind("a",b,c)	Zoekt in de string naar	"barabara".rfind("bar")
	laatste voorkomen van	
	string a vanaf b tot c	
"a".join(Lijst)	Plakt een lijst van	"".join(list)
	strings aaneen met a	
	als separator	
Sum(list, a)	Telt alle waarden van	Sum(lijst, 10)
	de list op met	
	startwaarde a	

List.append(a)	Voegt element a toe aan einde van list	Lijst.append("peer")
List.extend(a)	Voegt lijst a toe aan einde van list	Lijst.extend(["framboos", "aardbei", "aalbes"]) print(fruitlist)
List.insert(a, b)	Voegt element b toe op index a van list	Lijst.insert(3, "banaan")
List.remove(a)	Verwijdert de eerste instantie a in list, anders runtime error	Lijst.remove("appel")
List.pop(a)	Verwijdert element met index a in list en retourneert dit, anders runtime error	Lijst.pop(2)
List.del(a)	Verwijdert element of substring met index a zonder retourwaarde, anders runtime error	Del lijst[2:5]
List.clear()	Maakt list helemaal leeg	Lijst.clear()
Random.shuffle(list)	Ordent de elementen in een lijst random	Random.shuffle(lijst)
List.index(a, startwaarde, stopwaarde)	Retourneert de index van de eerste instantie van a in list, anders runtime error OOK VOOR TUPLES!	Lijst.index("ananas")
List.count(string, startwaarde, eindwaarde)	Telt hoeveel instanties er van de string in de list zijn.	['wortel', 'peer'].count('wortel')
List.sort()	Sorteert numerieke list chronologisch of stringlist alfabetisch	Lijst.sort()
List.sort(reverse=boolean)	Sorteert in omgekeerde volgorde	Lijst.sort(reverse=True)
List.sort(key=a)	Sorteert volgens de functie a	Lijst.sort(key=len)/Lijst.sort(key=str.lower)
Id(list)	Retourneert Id van de list	Id(lijst)
Is	Operator die ID's van lists vergelijkt	Lijst2 is Lijst1
Ord(a)	Geeft de ASCII- nummering van een teken	Ord("Z")
Chr(a)	Geeft het teken van de ASCII-nummering	Chr(55)
<handle> = open(bestandsnaam>)</handle>	Opent een bestand met een bepaalde handle om te lezen	Fp = open("test.txt")

open(<bestandsnaam>) as <handle></handle></bestandsnaam>		
<handle>.read()</handle>	Leest het bestand met de handle helemaal uit	Fp.read()
<handle>.close()</handle>	Sluit het bestand met de handle	Fp.write()
<handle>.readline()</handle>	Leest 1 lijn van het bestand met de handle uit	Fp.readline()
<handle>.readlines()</handle>	Retourneert een list met als elementen de lines van bestand met deze handle	F.readlines()
<handle> = open(bestandsnaam> , "w")</handle>	Opent een bestand met een bepaalde handle om te schrijven	Fp = open("test.txt", "w")
<handle> = open(bestandsnaam> , "a", encoding="latin-1")</handle>	Opent een bestand met een bepaalde handle om te appenden	Fp = open("test.txt", "a")
Os.path.exists(a)	Retourneert True als het pad a bestaat	exists("pc_rose.txt")
Os.path isfile(a)	Retourneert True als het pad a bestaat en a een bestand is	isfile("pc_rose.txt")
Os.path Isdir(a)	Retourneert True als het pad a bestaat en a een map is	isdir("pc_rose.txt")
Os.path Join(map, bestand)	Retourneert een path opgebouwd uit de meegegeven parameters	join(getcwd(), naam)
Os.listdir(a)	Retourneert een list met als elementen de bestanden in de map a	Listdir(.)
Os.path basename(a)	Retourneert de bestandsnaam uit path a	basename("/System/Home/test.txt")
Os.path Dirname(a)	Retourneert de mapnaam uit path a	dirname("/System/Home/test.txt")
Os.path Getsize(a)	Retourneert de bestandsgrootte in bytes van a	getsize("/System/Home/test.txt")
set.add(a)	Voegt het element a toe aan de set als het geen duplicaat is	Fruitset.add("appel")
set.update(a) set = a	Voegt de elementen van list a toe aan de set als het geen duplicaat is	fruitset.update(["appel","appel","aardbei", "aardbei","appel","mango"])

Set.remove(a) / set.discard(a)	Verwijdert element a	fruitset.remove("appel")
	uit de set (discard	
	geeft geen runtime	
	error indien element	
	niet bestaat)	
Set.pop()	Verwijdert random	fruitset.pop("appel")
	element uit de set en	
	retourneert het	
Set.union(a)	Retourneert de	fruit1.union(fruit2)
Set a	vereniging van de set	
	met set a	
Set.intersection(a)	Retourneert de	fruit1.intersection(fruit2)
Set & a	doorsnede van de set	
	met set a	
Set.difference(a)	Retourneert de	fruit1.difference(fruit2)
Set – a	elementen van de set	,
	die niet in a zitten	
Set.isdisjoint(a)	Retourneert True als	fruit1.isdisjoint(fruit2)
Set.isdisjonit(a)	geen enkel element	Truit1.13d13Johnt Truit2 /
	van de set in a zit	
Satissubsat(a)	Retourneert True als	fruit1 issubset/ fruit2 \
Set.issubset(a)	elk element van de set	fruit1.issubset(fruit2)
6	in a zit	(),4 (),2 ()
Set.issuperset(a)	Retourneert True als	fruit1.issuperset(fruit2)
	elk element van a in	
	de set zit	
Set.symmetric_difference(a)	Retourneert alle	fruit1.symmetricdiffernce(fruit2)
Set ^ a	elementen van de set	
	en a behalve de	
	doorsnede	
frozenset(a)	Creëert een	frozenset(["appel", "banaan", "kers"])
	onveranderbare set	
	met element(en) a	
Os.getcwd()	Geeft het path van de	Os.getcwd()
	huidige map	
Sys.Getfilesystemencoding()	Geeft de encoding van	Sys.Getfilesystemencoding()
sys. detinesystemeneoding()	het besturingssysteem	Sys. Getmesystementoum g()
	weer	
U = Urllib.request.urlopen(a)	Opent de website met	U = urlopen("http://www.python.org")
o – oriiib.request.urioperi(a)	link a met handle U	o - unopen(nttp.//www.python.org)
Urllib.error.HTTPError	Op te vangen	except HTTPError as e:
Urllib.error.URLError	exceptions bij openen	print("HTTP Error", e)
	van een webpagina	sys.exit()
		except URLError as e:
		print("URL error", e)
		sys.exit()
C = collections.Counter(data)	Counter object met	from collections import Counter
C most somman/	bijhorende functies	
C.most_common()	,	
C.most_common() C.update(iterable)		data = ["appel", "banaan", "appel",

	I	
		c = Counter(data)
		print(c)
		print(c.most_common(1))
		data2 = ["mango", "kers", "kers", "kers", "kers"]
		c.update(data2)
		print(c)
		print(c / print(c.most_common())
Dq = collections.deque(list)	Deque object met	from collections import deque
	bijkomende functies	· ·
Append(left)()		dq = deque([1, 2, 3])
Extend(left)()		dq.appendleft(4)
Pop(left)()		dq.extendleft([5, 6])
		print(dq)
Glist = glob.glob(regex)	Creëert list met alle	from glob import glob
	filenames die aan de	
	meegegeven regex	glist = glob("*.pdf")
	voldoen	for name in glist:
		print(name)
Glob.iglob(regex)	Creëert iterator voor	for name in glob.iglob("*.pdf"):
	alle filenames die aan	print(name)
	de meegegeven regex	
	voldoen	
Statistics.mean()	De statistics module	from statistics import mean, median,
Statistics.median()	met zijn functies	mode, stdev, variance, \
Statistics.(multi)mode()		StatisticsError
Statistics.stdev()		
Statistics.variance()		data = [4, 5, 1, 1, 2, 2, 2, 3, 3, 3]
		print("gemiddelde:", mean(data))
		print("mediaan:", median(data))
		try:
		print("modus:", mode(data))
		except StatisticsError as e:
		print(e)
		<pre>print("st.dev.: {:.3f}".format(stdev(data)</pre>
))
		<pre>print("variantie: {:.3f}".format(variance(</pre>
		data)))
Statistics.StatisticsError	Op te vangen	except StatisticsError as e:
	exception bij het	print(e)
	gebruik van statistics-	
	functions, bv. wanneer	
	er geen unieke modus	
	is	
Tijd =datetime.datetime(year,	De datetime module	ditjaar = datetime.now().year
month, day, uur = 0, minute =	en zijn 2 classes	xmasditjaar = datetime(ditjaar, 12, 25,
0, second = 0, microsecond =		23, 59, 59)
0, tzinfo = 0)		dezedag = datetime.now()
		dagen = xmasditjaar - dezedag

Delta =	
datetime.timedelta(day = 0,	if dagen.days < 0:
second = 0, microsecond = 0)	print("Kerst komt volgend jaar weer.")
	elif dagen.days == 0:
	print("Het is Kerst!")
	else:
	print("Slechts", dagen.days, "dagen tot
	Kerst!")

Conditie/iteratie:

- → If (+else/elif)
- → While (+else)
- → For (+else)

Exceptions:

- ZeroDivisionError:
 - o Delen door nul
- IndexError
 - Het benaderen van een list of tuple met een index die niet binnen het legale bereik valt
- NameError
 - o Onbekende variabelenaam
- KeyError
 - o Het benaderen van een dictionary met een key die onbekend is
- IOError
 - ledere fout die kan optreden als je een bestand benadert (deze exception is een alias voor OSError)
- FileNotFoundError
 - o Het proberen te openen van een niet-bestaand bestand om eruit te lezen
- ValueError
 - o Het optreden van een fout bij het "casten" van een waarde naar een andere waarde
- TypeError
 - Het gebruiken bij een operatie van een waarde met een data type dat niet ondersteund wordt door de operatie
- errno.ENOENT
 - Dit bestand of deze directory bestaat niet. Dit krijg je als je een bestand benadert dat niet bestaat.
- errno.EACCESS
 - Toestemming geweigerd. Je kunt deze melding in verschillende omstandigheden krijgen, zoals wanneer je probeert te lezen uit een gesloten bestand, of als je in een bestand probeert te schrijven dat voor alleen lezen bedoeld is, of als je een directory probeert te openen alsof het een bestand is.

• errno.ENOSPC

 Onvoldoende ruimte. Je krijgt deze fout als je een bestand probeert te schrijven en er geen ruimte voor het bestand beschikbaar is, bijvoorbeeld als je probeert te schrijven naar een USB-stick die vol is.

Termen:

- Text Editor ← → Word Processor
- Pure functies ←→ Modifiers
- \circ Formele parameter $\leftarrow \rightarrow$ Actuele parameter