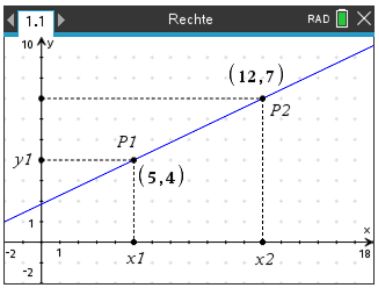
1. Inleidend voorbeeld

We tekenen een rechte in de TI-Nspire CX Graphs App door de punten $P1$ en $P2$.



We bewaren de coördinaten van de punten $P1$ en $P2$ in de volgende TI-Nspire CX variabelen $x1, y1$ en $x2, y2$ als volgt:

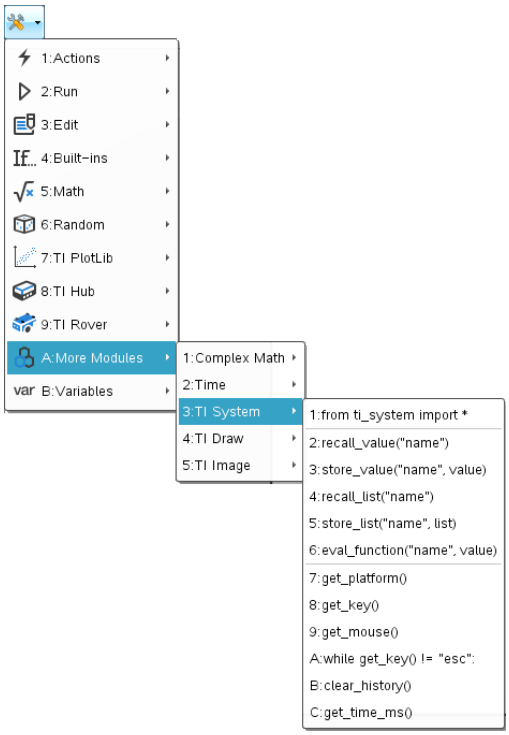
$P1\ (x1, y1)$ & $P2\ (x2, y2)$.

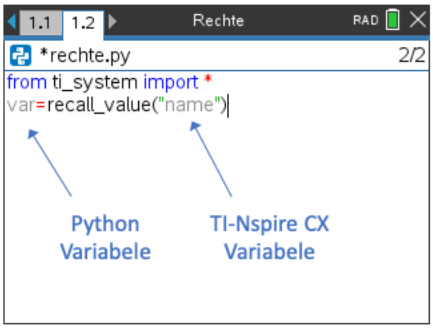
De vergelijking van de rechte is gelijk aan:

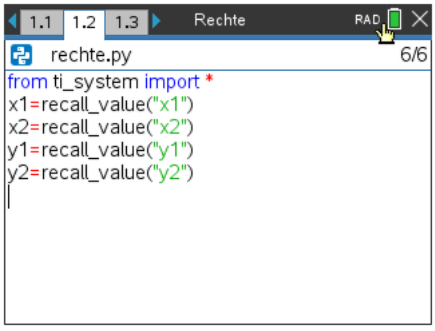
$$y - y1 = \frac{y2-y1}{x2-x1}(x - x1).$$

Met de module TI System kan variabelen vanuit TI-Nspire CX geïmporteerd worden in een Python-programma als Python variabelen. Na het importeren van de module ti_system selecteren we het commando recall_value().

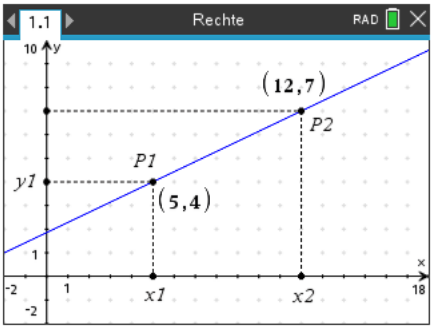
De onderstaande syntax-template verschijnt die aangeeft hoe de Python variabele te declareren op basis van de TI-Nspire CX variabele.

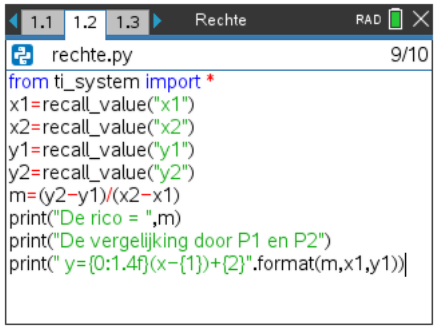




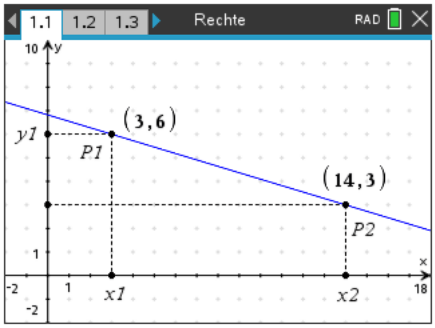


We berekenen de richtingscoëfficiënt m en bepalen de rechte door de twee punten $P1$ en $P2$.

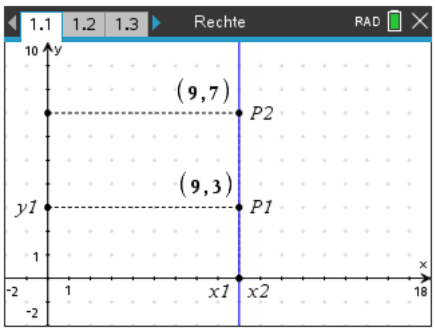


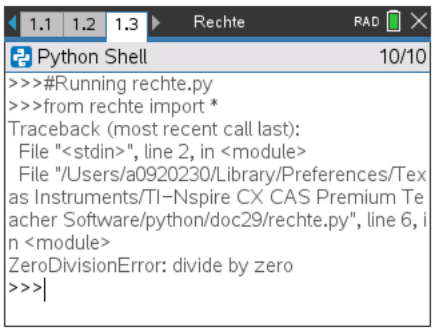


Als we de punten verplaatsen en het programma opnieuw runnen krijgen we de nieuwe vergelijking:



Maar wat met een verticale rechte?





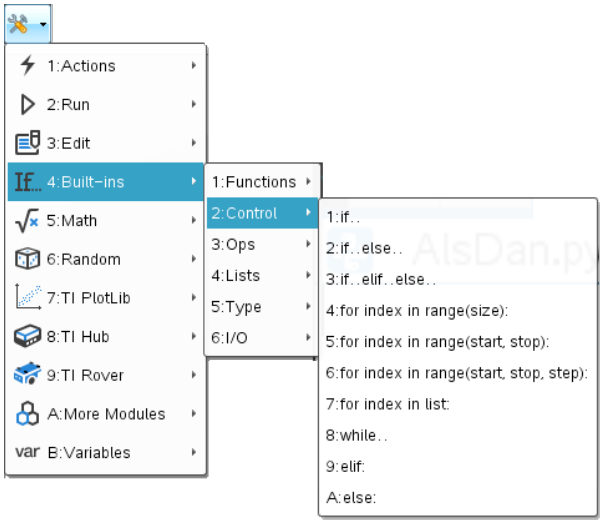
2. If-Statements

Om bovenstaande foutmelding te voorkomen testen we de conditie x1 == x2 of x1 != x2.

Hiervoor selecteren we in het Control-menu de optie 2: if..else.. en passen ons programma als volgt aan.

Om commentaar in te voeren start je de regel met #.

De template van het if-statement verschijnt automatisch in de Program Editor.

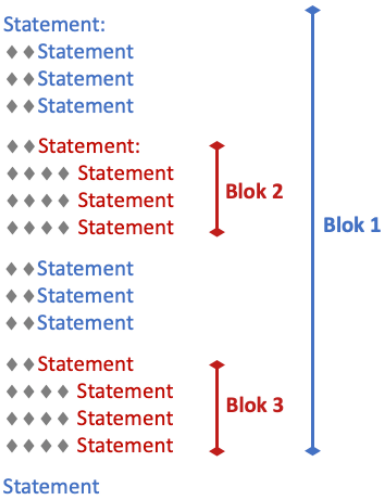


```
if BooleanExpr:
    block
else:
    block
```

Een if-statement bestaat uit het sleutelwoord if, gevolgd door een voorwaarde en een dubbelpunt. De statements na het dubbelpunt, moeten in een blok staan. Indien de voorwaarde True is, wordt alle statements in het blok uitgevoerd.

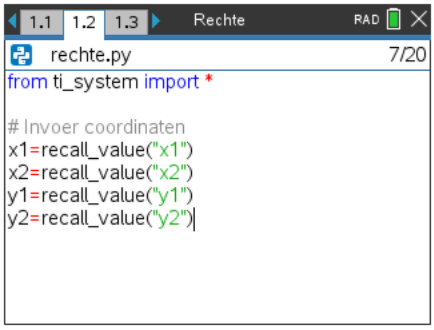
Een blok is een verzameling van gegroepeerde statements. De Whitespace voor de statements geef aan welke statements tot eenzelfde blok. Statements die op dezelfde positie starten, m.a.w. regels met dezelfde inspringingen, behoren tot hetzelfde blok.

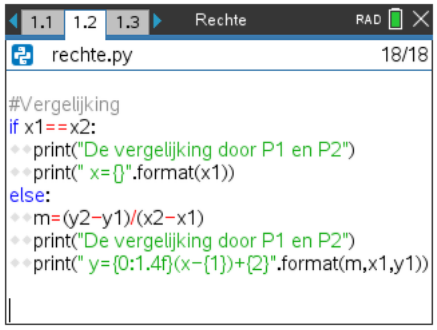
Merk op dat if-statements niet afgesloten worden met een End-statement maar begin en einde is gebaseerd op gelijke inspringingen. In TI-Technologie wordt de Whitespace aangeduid met ♦♦.



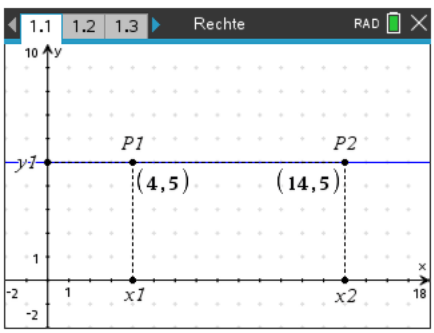
Een if-statement kan uitgebreid worden met een else-statement met eventueel een elif-statement tussenin.

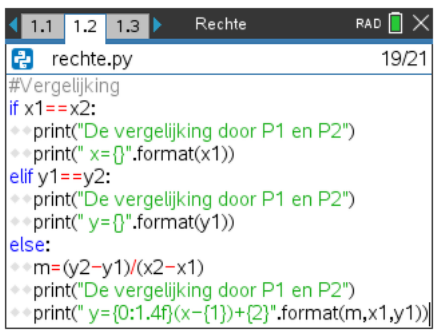
Voor ons programma geeft dit het volgende:





Met een extra elif-statement kunnen we ook de vergelijking van een horizontale rechte eleganter weergeven:

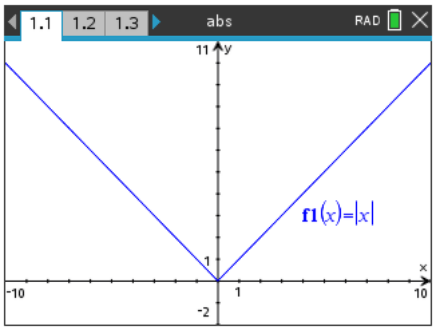


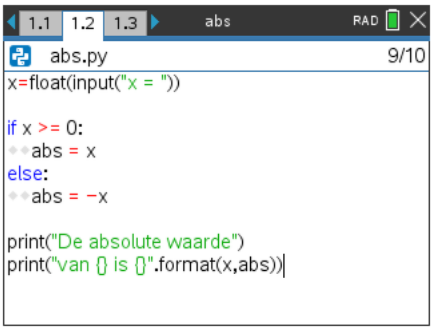


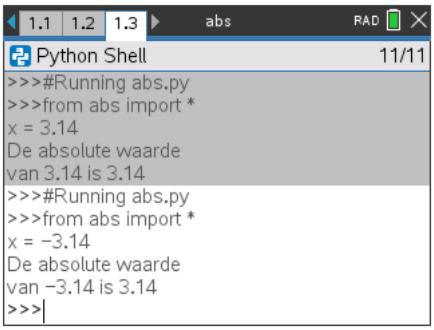
Om de structuur van Python-code goed op een rij te hebben, is het belangrijk steeds aandacht te hebben op hoe de structuur van een Python-programma gebaseerd is op inspringingen (indents of whitespace).

Nog twee voorbeelden om de structuur van de if-statements te illustreren.

1. Bepaal de absolute waarde van een reëel getal baserend op de definitie $|x| = \begin{cases} x \ als\ x \leq 0 \\ -x\ als\ x < 0 \end{cases}$.







2. Bereken de functiewaarde voor de volgende stuksgewijs gedefinieerde functie $f1(x) = \begin{cases} x+2\ als\ x \leq -2 \\ 0\ als -2 < x < 2 \\ x-2\ als\ x \geq 2 \end{cases}$.