



# LIBRARY TURTLE



## Modul Pembelajaran Untuk Pemrograman Logo

Tim Penyusun:

Dr. Harja Santana Purba, M.Kom.

Rizky Pamuji, S.Kom., M.Kom.

Khoyrur Roykhan



## KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya sehingga penyusunan materi ini dapat diselesaikan dengan baik. Materi ini disusun untuk membantu siswa dalam memahami dasar-dasar pemrograman melalui visualisasi yang interaktif dan menyenangkan.

Library Turtle di Python merupakan salah satu modul standar yang dirancang untuk memudahkan pengenalan konsep pemrograman melalui grafik. Modul ini terinspirasi dari bahasa Logo, yang diperkenalkan oleh Wally Feurzeig, Seymour Papert, dan Cynthia Solomon pada tahun 1967. Dengan Turtle, pengguna dapat membuat berbagai bentuk geometris dan pola dengan mengendalikan pergerakan "pen" atau "kura-kura" di atas layar, sehingga sangat cocok untuk pemula yang ingin mempelajari konsep dasar pemrograman, matematika, dan geometri.

Dalam penyusunan materi ini, konsep Turtle dimodifikasi dengan menghadirkan karakter Bidawang sebagai pengganti "kura-kura." Bidawang, yang dikenal dalam kearifan lokal sebagai hewan khas lahan basah, diharapkan dapat memberikan pengalaman belajar yang lebih kontekstual dan relevan bagi siswa, terutama dalam mengenalkan potensi dan lingkungan sekitar.

Akhir kata, penulis berharap materi ini dapat memberikan manfaat dan menjadi sarana pembelajaran yang efektif. Segala kritik dan saran yang membangun sangat diharapkan untuk perbaikan di masa mendatang.

*Penyusun*

## DAFTAR ISI

<b>1. PENGENALAN .....</b>	<b>6</b>
<b>2. PERGERAKAN .....</b>	<b>10</b>
A. Left & Right .....	10
B. Forward & Backward.....	13
C. Setposition .....	16
D. Setx & Sety.....	19
E. Setheading .....	22
F. Home.....	25
G. Circle.....	28
H. Dot .....	31
<b>3. MENGETAHUI STATUS .....</b>	<b>35</b>
A. Position .....	35
B. Xcor & Ycor .....	39
C. Heading.....	43
D. Distance.....	46
<b>4. KONTROL PENA .....</b>	<b>49</b>
A. Pendown & Penup .....	49
B. Pense .....	52
C. Isdown .....	55
<b>5. KONTROL WARNA.....</b>	<b>58</b>
A. Pencolor .....	58
B. Fillcolor, Begin_fill, & End_fill .....	61
<b>6. KONTROL GAMBAR LANJUTAN .....</b>	<b>64</b>
A. Reset.....	64
B. Clear.....	67
C. Write.....	70

## DAFTAR GAMBAR

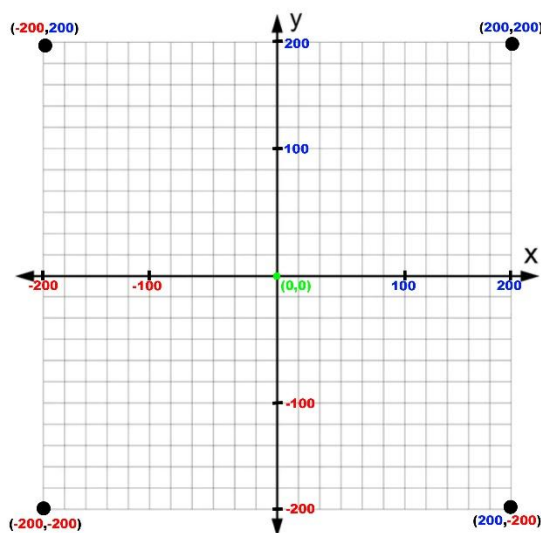
## DAFTAR TABEL

## Tujuan Pembelajaran

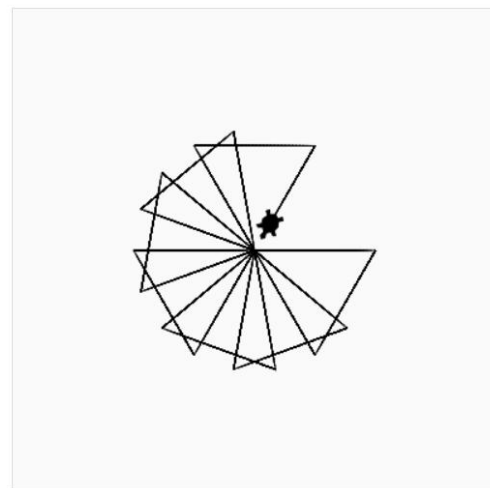
- Memahami konsep canvas sebagai ruang pergerakan Bidawang.
- Mengenali tampilan lingkungan kerja dan perintah dasar untuk menggerakkan Bidawang pada canvas.

## Canvas

Canvas adalah area tempat Bidawang bergerak menggambar pola geometri secara interaktif. Bidawang tersebut dapat dikontrol untuk bergerak maju (forward), mundur (backward), berbelok ke kiri (left), berbelok ke kanan (right), dan melakukan berbagai aksi lainnya menggunakan perintah-perintah tertentu. Ukuran default canvas adalah 400x400 piksel. Berikut adalah contoh gambaran canvas dan contoh Bidawang bergerak dalam canvas:



Gambar 1 (Gambaran koordinat canvas)



Gambar 2 (Contoh pergerakan Bidawang di canvas)

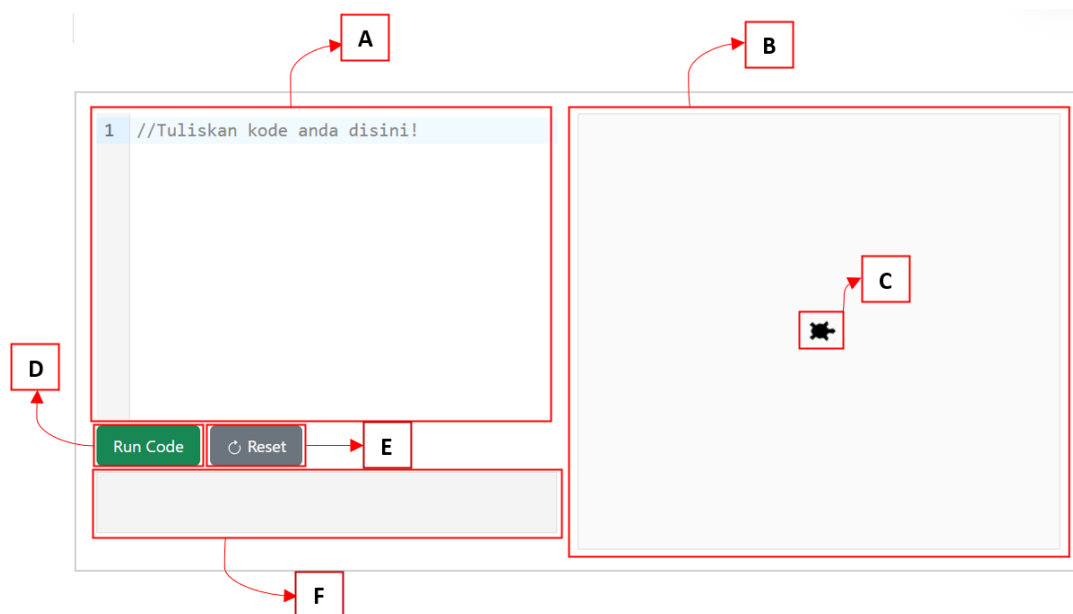
## Penjelasan:

- Titik awal posisi Bidawang adalah (0, 0), yang berada di tengah canvas.
- Batas pergerakan ke atas (sumbu Y positif) adalah 200, yang merupakan batas atas canvas.

- Batas pergerakan ke bawah (sumbu Y negatif) adalah -200, yang merupakan batas bawah canvas.
- Batas pergerakan ke kanan (sumbu X positif) adalah 200.
- Batas pergerakan ke kiri (sumbu X negatif) adalah -200.

## Tampilan Lingkungan Kerja

Untuk mempermudah mengontrol Bidawang, tersedia lingkungan kerja di dalam aplikasi yang sudah di kembangkan, yang terdiri dari beberapa komponen seperti gambar dibawah ini:



Gambar 3 (Tampilan lingkungan kerja)

Penjelasan :

- **(A) Text Editor** : Area tempat pengguna mengetik perintah kode untuk menggerakkan Bidawang.
- **(B) Canvas** : Area tampilan di mana pergerakan Bidawang divisualisasikan. Semua perintah yang dijalankan akan langsung terlihat pada canvas.
- **(C) Bidawang** : Objek yang digerakkan menggunakan perintah kode.

- **(D) Tombol "Run Code"**: Digunakan untuk menjalankan kode yang telah ditulis di text editor. Setelah ditekan, Bidawang akan menjalankan perintah dan menggambar sesuai instruksi.
- **(E) Tombol "Reset"**: Menghapus kode serta hasil gambar di canvas dan mengembalikan Bidawang ke posisi awal.
- **(F) Output Log**: Digunakan untuk menampilkan output dari program yang dijalankan atau pesan error.

## Aktivitas 1: Menggerakkan Bidawang di Dalam Canvas

Yuk, kita belajar cara menggerakkan Bidawang dengan berbagai perintah dasar! Ikuti langkah-langkah di bawah ini dengan membuka aplikasi editor yang sudah disiapkan. Ketikkan setiap perintah satu per satu dan tekan tombol **"Run Code"** untuk melihat Bidawang bergerak di canvas.

### Langkah-langkah:

#### 1. Maju

Gerakkan Bidawang maju sejauh 100 langkah dengan perintah dibawah ini:

```
forward 100
```

#### 2. Berbelok ke kanan

Kemudian lanjutkan lagi pada baris baru dengan perintah dibawah ini untuk memutar Bidawang ke kanan sebesar 90 derajat:

```
right 90
```

#### 3. Maju

Gerakkan lagi Bidawang maju sejauh 100 langkah:

```
forward 100
```

#### 4. Berbelok ke kiri

Putar Bidawang ke kiri sebesar 45 derajat:

```
left 45
```



## 5. Maju

Gerakkan Bidawang maju sejauh 50 langkah:

forward 50

## Kuis

1. Jika pergerakan Bidawang ke kanan adalah 200 dan ke bawah adalah -150, bagaimana koordinat tersebut dijelaskan?
  - a. (200, 150)
  - b. (-200, 150)
  - c. (200, -150)
  - d. (-200, -250)

### A. Left & Right

#### Tujuan Pembelajaran

- Memahami cara mengendalikan arah rotasi Bidawang menggunakan left dan right.

Perintah **left** dan **right** digunakan untuk memutar arah gerakan Bidawang berdasarkan sudut derajat yang diberikan, tanpa harus memindahkan posisinya. Ini berguna untuk mengatur arah Bidawang sebelum melanjutkan dengan perintah lainnya seperti bergerak.

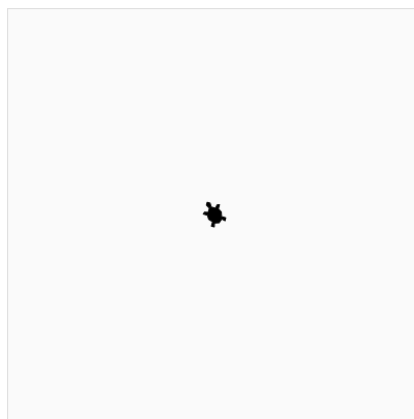
#### 1. Left

Digunakan untuk memutar arah Bidawang berlawanan arah jarum jam (kiri) sebesar derajat yang ditentukan.

Contoh :

```
left 120
```

Hasil :



Gambar 4 (Hasil contoh penggunaan left)

Bidawang yang awalnya menghadap ke kanan layar, akan berputar 120 derajat ke kiri.

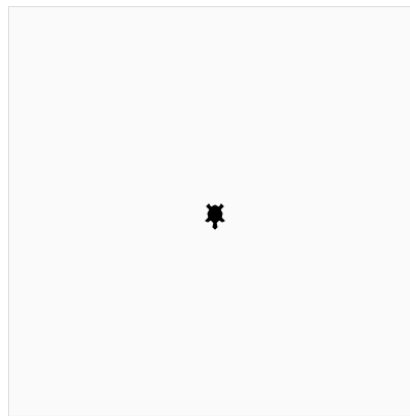
## 2. Right

Digunakan untuk memutar arah Bidawang searah jarum jam (kanan) sebesar derajat yang ditentukan.

Contoh:

```
right 90
```

Hasil:



Gambar 5 (Hasil contoh penggunaan right)

Bidawang yang awalnya menghadap ke kanan layar, akan berputar 90 derajat ke kanan.

### Aktivitas 2.1 : Mengenal Perintah left dan right

Sekarang saatnya kamu berlatih memahami bagaimana **Bidawang berputar** ke arah kiri dan kanan. Buka aplikasi yang telah disiapkan, lalu ikuti instruksi di bawah ini. Ketikkan perintah satu per satu dan tekan tombol "**Run Code**" untuk melihat hasilnya di canvas:

#### Langkah-langkah:

1. Berputar ke kiri

Buat bidawang berputar 90 derajat ke kiri dengan perintah di bawah ini:

```
left 90
```

## 2. Berputar ke kanan

Kemudian lanjutkan pada baris baru dengan perintah dibawah ini untuk memutar Bidawang ke kanan sebesar 180 derajat:

```
right 180
```

## Kesimpulan

Perintah ``left`` dan ``right`` memungkinkan pengaturan arah gerakan dengan rotasi ke kiri atau ke kanan berdasarkan derajat yang ditentukan. Perintah ini sangat berguna untuk kontrol arah sebelum melakukan perintah lain dalam pembuatan gambar atau pola.

## Pertanyaan

1. Apa perbedaan utama antara perintah `left` dan `right` ?
  - a. `left` memutar bidawang ke arah kanan, sementara `right` memutar bidawang ke arah kiri.
  - b. `left` memutar bidawang ke arah kiri, sementara `right` memutar bidawang ke arah kanan.
  - c. `left` dan `right` hanya digunakan untuk mengubah warna turtle.
  - d. Keduanya memindahkan turtle ke posisi (0, 0).
2. Jika bidawang menghadap ke timur (kanan layar), dan Anda menggunakan perintah `right 90`, ke arah mana bidawang akan menghadap?
  - a. Utara (atas layar)
  - b. Barat (kiri layar)
  - c. Selatan (bawah layar)
  - d. Timur (kanan layar)

## B. Forward & Backward

### Tujuan Pembelajaran

- Memahami cara menggerakkan Bidawang ke depan dan ke belakang menggunakan forward dan backward.

Perintah `forward` dan `backward` digunakan untuk menggerakkan Bidawang ke arah depan (forward) searah arah yang sedang dihadapi Bidawang dan belakang (backward) berlawanan arah dengan yang sedang dihadapi Bidawang, berdasarkan jarak yang ditentukan.

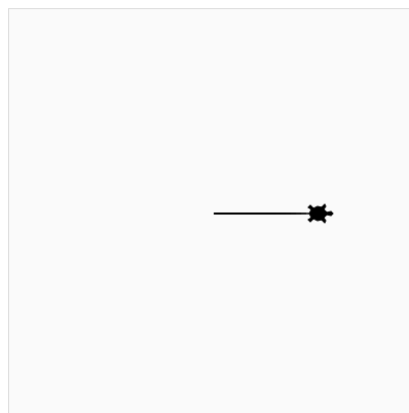
#### 1. Forward

Digunakan untuk menggerakkan Bidawang ke depan sejauh jarak yang ditentukan.

Contoh:

```
forward 100
```

Hasil:



Gambar 6 (Hasil contoh penggunaan forward)

Bidawang akan bergerak sejauh 100 langkah ke arah yang dihadapnya.

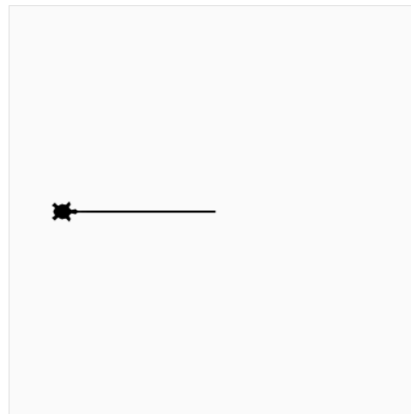
## 2. Backward

Menggerakkan Bidawang ke belakang sejauh jarak yang ditentukan (dalam piksel), dalam arah berlawanan dengan arah yang sedang dihadapi oleh bidawang.

Contoh:

```
backward 150
```

Hasil:



*Gambar 7 (Hasil contoh penggunaan backward)*

Bidawang akan mundur sejauh 150 langkah dari arah yang dihadapnya.

## Aktivitas 2.2 : Memahami Gerakan Maju dan Mundur

Saatnya latihan menggerakkan Bidawang ke **depan dan ke belakang**! Buka aplikasi yang telah disiapkan, lalu ikuti setiap langkah berikut ini. Ketik perintah satu per satu, kemudian klik tombol "**Run Code**" untuk melihat pergerakannya langsung di canvas.

### Langkah-langkah:

#### 1. Maju

Gerakkan Bidawang maju sejauh 100 langkah dengan perintah dibawah ini:

```
forward 100
```

#### 2. Mundur

Kemudian lanjutkan lagi pada baris baru dengan perintah dibawah ini untuk membuat Bidawang mundur 50 langkah:

```
backward 50
```

3. Mundur

Gerakkan lagi bidawang mundur sejauh 100 langkah:

```
backward 100
```

## Kesimpulan

Perintah forward dan backward digunakan untuk menggerakkan Bidawang ke depan atau ke belakang sejauh jarak yang ditentukan dalam piksel. Perintah forward dan backward sering dikombinasikan dengan perintah rotasi seperti `left` dan `right` untuk membuat pola atau gambar yang lebih kompleks.

## Pertanyaan

1. Perintah apa yang digunakan untuk menggerakkan turtle ke depan sejauh 150 piksel?
  - a. backward(150)
  - b. forward(150)
  - c. left(150)
  - d. right(150)
2. Jika turtle menghadap ke barat dan Anda menggunakan perintah backward(100), ke arah mana turtle akan bergerak?
  - a. Ke barat
  - b. Ke timur
  - c. Ke utara
  - d. Ke Selatan

## C. Setposition

### Tujuan Pembelajaran

- Memahami cara mengatur posisi Bidawang secara langsung menggunakan ``setposition``.

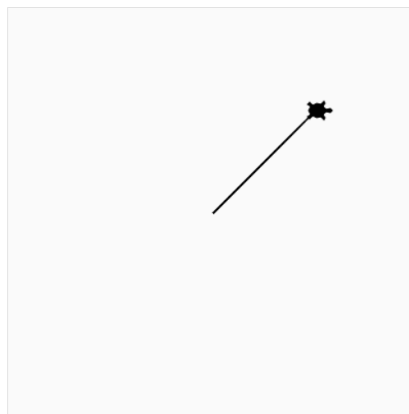
Perintah ``setposition`` (bisa juga disingkat ``setpos``) digunakan untuk memindahkan Bidawang ke titik tertentu dalam ruang canvas, berdasarkan koordinat yang ditentukan. Saat perintah ini digunakan, Bidawang akan menggambar jalur dari posisi sebelumnya ke posisi baru.

### Contoh:

Memindahkan Bidawang ke koordinat ( $x = 100$ ,  $y = 100$ ) dalam ruang canvas:

```
setposition 100 100
```

Hasil:



Gambar 1 (Hasil contoh penggunaan `setposition`)

Bidawang akan bergerak dari posisi awal ( $x=0$ ,  $y=0$ ) ke posisi  $x=100$  dan  $y=100$ .



## Aktivitas 2.3: Mengenal Perintah `setposition`

Sekarang kita akan belajar bagaimana memindahkan Bidawang langsung ke titik tertentu di canvas. Perintah `setposition` memungkinkan Bidawang **melompat ke posisi baru** tanpa perlu berputar atau melangkah perlahan.

Buka aplikasi yang telah disiapkan, lalu ikuti instruksi berikut ini. Ketikkan setiap perintah satu per satu dan tekan tombol **"Run Code"** untuk melihat hasilnya.

### Langkah-langkah:

#### 1. Berpindah posisi

Ubah posisi bidawang yang berada di titik awal menjadi ke titik (100,100) dengan perintah dibawah ini:

```
setposition 100 100
```

#### 2. Berpindah posisi

Lanjutkan pada baris baru untuk mengubah posisi bidawang lagi ke titik (0,100) dengan perintah dibawah ini:

```
setposition 0 100
```

## Kesimpulan

Perintah ``setposition`` atau ``setpos`` digunakan untuk memindahkan Bidawang ke koordinat tertentu dalam ruang canvas. Perintah ini sangat berguna untuk mengatur posisi awal Bidawang atau membuat jalur menggambar yang kompleks dengan mengombinasikan pergerakan dan kontrol pena.

## Pertanyaan

1. Apa fungsi utama dari perintah `setposition(x, y)`?
  - a. Mengatur warna garis.
  - b. Memutar turtle ke arah tertentu.
  - c. Memindahkan bidawang ke koordinat tertentu dalam ruang gambar.
  - d. Mengubah ukuran turtle.
  
2. Jika posisi awal Bidawang adalah (0, 0), dan Anda menggunakan perintah `setposition(50, 100)`, di mana posisi akhir Bidawang?
  - a. (100, 50)
  - b. (50, 0)
  - c. (0, 100)
  - d. (50, 100)

## D. Setx & Sety

### Tujuan Pembelajaran

- Memahami cara mengatur posisi bidawang pada sumbu x dan y secara terpisah menggunakan `setx` dan `sety`.

Fungsi setx dan sety digunakan untuk memindahkan Bidawang ke posisi tertentu hanya pada sumbu x (setx) atau sumbu y (sety), tanpa mengubah koordinat lainnya. Perintah ini berguna untuk memindahkan bidawang secara horizontal atau vertikal.

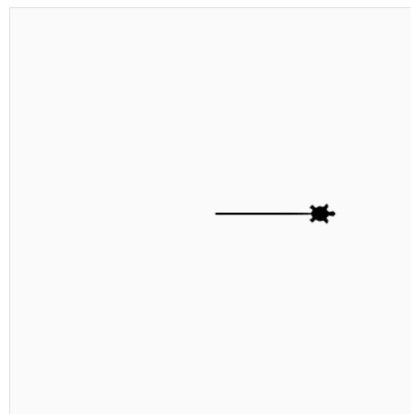
#### 1. Setx

Digunakan untuk memindahkan Bidawang ke posisi x yang ditentukan, tanpa mengubah posisi pada sumbu y.

Contoh:

```
setx 100
```

Hasil:



Gambar 2 (Hasil contoh penggunaan setx)

Posisi awal Bidawang adalah  $x=0, y=0$ . Setelah dijalankan kode tersebut maka nilai x akan menjadi 100. Dan Bidawang akan berpindah ke titik  $x=100, y=0$ .

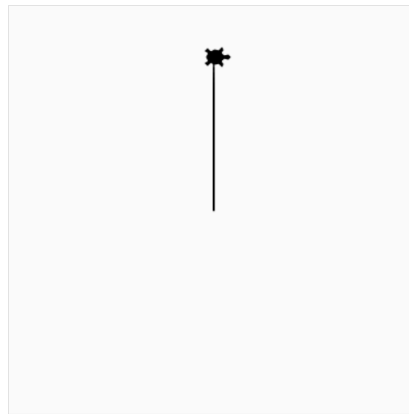
## 2. Sety

Digunakan untuk memindahkan Bidawang ke posisi y yang ditentukan, tanpa mengubah posisi pada sumbu x.

Contoh:

```
sety 150
```

Hasil:



*Gambar 3 (Hasil contoh penggunaan sety)*

Posisi awal bidawang adalah  $x=0$ ,  $y=0$ . Setelah dijalankan kode tersebut maka nilai y akan menjadi 150. Dan bidawang akan berpindah ke titik  $x=0$ ,  $y=150$ .

### Aktivitas 2.4: Mengubah Posisi Bidawang dengan setx dan sety

Kali ini kita akan belajar cara mengubah posisi Bidawang secara horizontal dan vertikal menggunakan perintah setx dan sety. Buka aplikasi yang sudah disiapkan, lalu ikuti instruksi di bawah ini. Ketikkan setiap perintah satu per satu dan tekan **"Run Code"** untuk melihat hasil gerakan di canvas.

Langkah-langkah:

#### 1. Ubah titik X

Ubah nilai titik **X** bidawang yang awalnya berada di titik awal (0,0) ubah titik **X** nya menjadi 100 dengan perintah dibawah ini:

```
setx 100
```

#### 2. Ubah titik Y

Kemudian lanjutkan pada baris baru untuk mengubah nilai titik **Y** menjadi -50 dengan perintah dibawah ini:

```
sety -50
```

3. Ubah titik X

Lalu ubah lagi nilai titik **X** menjadi 0 dengan perintah dibawah ini:

```
setx 0
```

## Kesimpulan

Perintah `setx` dan `sety` digunakan untuk memindahkan Bidawang secara horizontal atau vertikal tanpa mengubah koordinat lainnya. Perintah ini berguna untuk memindahkan Bidawang ke posisi yang diinginkan dengan lebih fleksibel, terutama saat mengatur pola atau jalur yang spesifik.

## Pertanyaan

1. Apa hasil dari perintah `setx 200` jika posisi awal turtle adalah (100,50)?
  - a. Posisi baru menjadi (200, 50).
  - b. Posisi baru menjadi (100, 200).
  - c. Posisi baru menjadi (200, 200).
  - d. Posisi baru tetap (100, 50).
2. Jika Anda ingin memindahkan turtle secara vertikal ke posisi  $y = -50$  tanpa mengubah posisi  $x$ , perintah apa yang harus digunakan?
  - a. `sety -50`
  - b. `setx -50`
  - c. `setposition -50 0`
  - d. `home`

## E. Setheading

### Tujuan Pembelajaran

- Memahami cara mengatur arah Bidawang menggunakan setheading.

Perintah setheading digunakan untuk mengatur arah Bidawang ke sudut tertentu, diukur dalam derajat. Sudut dihitung berlawanan arah jarum jam, dimulai dari arah timur (0 derajat). Dengan menggunakan setheading dapat mengontrol ke mana bidawang akan menghadap sebelum bergerak.

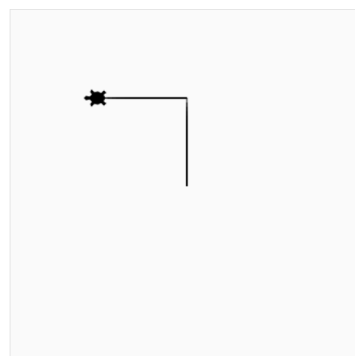
- **0°** : Timur
- **90°** : Utara
- **180°** : Barat
- **270°** : Selatan

### Contoh

Mengatur arah pergerakan bidawang dengan setheading:

```
setheading 90  
forward 100  
  
setheading 180  
forward 100
```

Hasil:



Gambar 4 (Hasil contoh penggunaan setheading)

## Aktivitas 2.5: Mengubah Arah Bidawang dengan setheading

Saatnya belajar bagaimana mengubah arah hadap Bidawang tanpa perlu memutar bertahap menggunakan left atau right. Buka aplikasi yang sudah disiapkan, lalu ikuti langkah-langkah di bawah ini. Ketikkan perintah satu per satu dan klik tombol **"Run Code"** untuk melihat pergerakannya di canvas.

### Langkah-langkah:

#### 1. Hadap Utara

Ubah arah bidawang menjadi menghadap ke arah utara dengan menggunakan perintah dibawah ini:

```
setheading 90
```

#### 2. Maju

Lalu lanjutkan pada baris baru untuk membuat bidawang maju sejauh 100 langkah dengan perintah dibawah ini:

```
forward 100
```

#### 3. Hadap Timur

Lanjutkan lagi dengan mengubah arah bidawang menjadi menghadap ke timur dengan perintah dibawah ini:

```
setheading 0
```

#### 4. Maju

Buat bidawang maju sejauh 50 langkah lagi.

```
forward 50
```

## Kesimpulan

Perintah setheading() sangat berguna untuk mengontrol arah objek dengan presisi. Dengan mengatur sudut arah secara langsung, Anda dapat membuat pola yang kompleks dan menggambar dengan lebih terstruktur.

## Pertanyaan

1. Jika Anda menggunakan perintah `setheading 90`, ke arah mana Bidawang akan menghadap?
  - a. Timur
  - b. Barat
  - c. Utara
  - d. Selatan
2. Apa fungsi utama dari perintah `setheading`?
  - a. Mengatur posisi turtle ke koordinat (0, 0).
  - b. Mengatur arah turtle ke sudut tertentu berdasarkan derajat.
  - c. Menggerakkan turtle ke arah utara.
  - d. Mengubah warna turtle.



## F. Home

### Tujuan Pembelajaran

- Memahami cara mengembalikan posisi Bidawang ke titik awal menggunakan home.

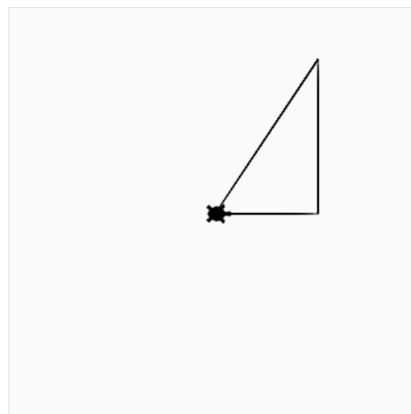
Perintah `home` digunakan untuk memindahkan Bidawang kembali ke posisi awalnya, yaitu titik (0, 0). Selain memindahkan Bidawang ke posisi awal, perintah ini juga mengatur arah Bidawang menghadap ke timur (0 derajat). Ini berguna ketika Anda ingin memulai kembali menggambar dari posisi awal.

### Contoh

Memindahkan bidawang ke posisi awal setelah menjalankan berbagai perintah:

```
forward 100  
left 90  
forward 150  
  
home
```

Hasil:



Gambar 5 (Hasil contoh penggunaan home)

Fungsi home akan membuat Bidawang kembali ke posisi awal setelah menjalankan perintah lainnya.

## Aktivitas 2.6: Kembali ke Titik Awal dengan home

Kita akan belajar bagaimana mengembalikan posisi dan arah Bidawang ke titik awal (0, 0) menggunakan perintah home. Buka aplikasi yang sudah disiapkan, lalu ikuti instruksi di bawah ini. Ketikkan setiap perintah secara berurutan dan tekan **"Run Code"** untuk melihat hasil pergerakannya.

### Langkah-langkah:

#### 1. Gerakkan Bidawang

Buat bidawang menjadi bergerak ke arah mana saja, sebagai contoh gunakan perintah di bawah ini untuk menggerakkan bidawang:

```
forward 100  
right 45  
forward 200
```

#### 2. Kembali ke posisi awal

Selanjutnya ketikkan perintah di bawah ini pada baris baru untuk mengembalikan bidawang ke posisi awal.

```
home
```

## Kesimpulan

Perintah home memudahkan untuk kembali ke posisi awal (0, 0) dan mengatur arah Bidawang ke posisi semula. Ini berguna untuk memulai kembali proses menggambar dari titik awal.

## Pertanyaan

1. Apa fungsi dari perintah home?
  - a. Menghapus seluruh gambar yang telah dibuat.
  - b. Mengembalikan Bidawang ke posisi awal (0, 0) dan mengatur arahnya ke timur.
  - c. Utara

- d. Memindahkan Bidawang ke posisi  $y = 0$ .
- 
- 2. Jika posisi awal Bidawang adalah (100, 100) dan arahnya ke barat, apa yang terjadi setelah menggunakan home?
    - a. Bidawang tetap di posisi (100, 100).
    - b. Bidawang kembali ke posisi (0, 0) dengan arah tetap ke barat.
    - c. Bidawang kembali ke posisi (0, 0) dan menghadap ke timur.
    - d. Bidawang tetap di posisi (100, 100) tetapi menghadap ke timur.

## G. Circle

### Tujuan Pembelajaran

- Memahami cara menggambar lingkaran menggunakan circle.

Fungsi circle digunakan untuk menggambar lingkaran atau bagian dari lingkaran (busur) dengan jari-jari tertentu. Fungsi ini sangat berguna untuk menggambar bentuk bulat dan pola yang melibatkan lingkaran.

circle(jari\_jari, extent=None): Menggambar lingkaran dengan jari-jari yang ditentukan. Parameter extent opsional dan digunakan untuk menggambar busur lingkaran. Jika extent tidak diberikan, maka lingkaran penuh akan digambar.

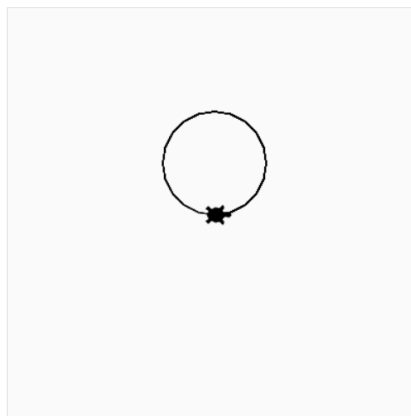
- **jari\_jari** : nilai jari-jari lingkaran yang ingin digambar.
- **Extent** : Sudut (dalam derajat) dari lingkaran yang ingin digambar. Jika tidak ada, menggambar lingkaran penuh (360 derajat).

### Contoh 1

Menggambar lingkaran penuh dengan menentukan jari-jari:

```
circle 50
```

Hasil:



Gambar 6 (Hasil contoh 1 penggunaan circle)

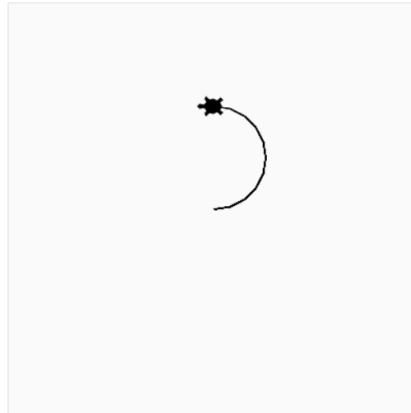
Fungsi circle 50 akan membuat Bidawang menggambar lingkaran dengan jari jari 50.

## Contoh 2

Menggambar busur dengan menentukan besaran derajatnya 'extent':

```
circle 50 180
```

Hasil:



Gambar 7 (Hasil contoh 2 penggunaan circle)

Fungsi `circle 50 180` akan membuat Bidawang menggambar lingkaran dengan jari jari 50, dengan besar busur 180 derajat.

## Aktivitas 2.7: Menggambar Lingkaran dan Busur

Sekarang saatnya belajar cara membuat Bidawang menggambar bentuk lingkaran penuh dan busur menggunakan perintah circle. Buka aplikasi yang sudah disiapkan, lalu ikuti instruksi di bawah ini secara bertahap. Ketikkan perintahnya dan klik **"Run Code"** untuk melihat hasilnya di canvas.

### Langkah-langkah:

1. Menggambar Lingkaran penuh

Buat bidawang menggambar lingkaran penuh dengan jari-jari 100 menggunakan perintah dibawah ini:

```
circle 100
```

2. Berpindah posisi

Kemudian pada baris berikutnya gunakan perintah `setposition` untuk memindahkan posisi bidawang ke (0,-200):

```
setposition 0 -200
```

### 3. Menggambar busur

Lalu lanjutkan lagi menggambar busur dengan besar 180 derajat dan jari jari 100 menggunakan perintah dibawah ini:

```
circle 100 180
```

## Kesimpulan

Perintah **circle** digunakan untuk menggambar lingkaran atau busur dengan mudah. Dengan mengatur jari-jari dan opsi ``extent``, kita dapat menggambar berbagai bentuk bulat yang berbeda untuk memperkaya gambar atau pola yang dibuat.

## Pertanyaan

1. Apa fungsi dari parameter `extent` dalam metode `circle`?
  - a. Mengatur arah lingkaran.
  - b. Menentukan ukuran jari-jari lingkaran.
  - c. Menentukan besaran sudut busur yang ingin Digambar.
  - d. Mengubah warna lingkaran.
2. Apa yang terjadi jika Anda menjalankan perintah `circle -50`?
  - a. Tidak ada lingkaran yang Digambar.
  - b. Lingkaran dengan jari-jari 50 akan digambar berlawanan dengan arah.
  - c. Lingkaran dengan jari-jari 50 akan digambar dengan arah normal (berlawanan arah jarum jam)
  - d. Lingkaran kecil akan digambar di posisi saat ini.

## H. Dot

### Tujuan Pembelajaran

- Memahami cara menggambar titik menggunakan dot.

Fungsi dot digunakan untuk menggambar titik pada posisi Bidawang berada. Titik ini dapat diatur ukuran dan warnanya. Fungsi ini sangat berguna untuk menandai posisi tertentu dalam gambar atau untuk menggambar pola dengan titik.

**dot(ukuran, warna):** Menggambar titik dengan ukuran dan warna yang ditentukan. Parameter warna opsional dan dapat diisi dengan nama warna atau kode heksadesimal.

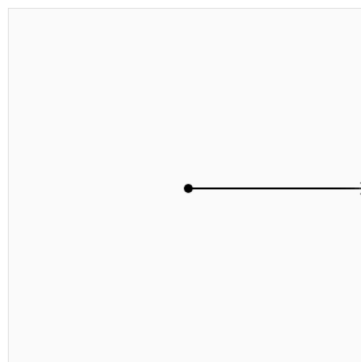
- **Ukuran** : Ukuran titik yang ingin digambar. Nilai default adalah 5.
- **Warna** : Warna titik yang ingin digambar. Jika tidak ditentukan, warna default adalah warna pena saat ini.

### Contoh 1

Menggambar titik dengan ukuran 10 dan warna default:

```
dot 10  
forward 200
```

Hasil:



Gambar 8 (Hasil contoh 1 penggunaan dot)

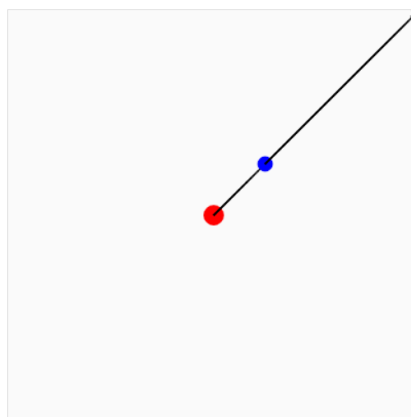
Fungsi `dot 10` akan menggambar titik pada posisi bidawang tersebut dengan ukuran 10 dan warna default.

## Contoh 2

Menggambar Titik dengan Ukuran dan Warna yang Berbeda:

```
dot 20 "red"  
setposition 50 50  
dot 15 "blue"  
setposition 200 200
```

Hasil:



Gambar 9 (Hasil contoh 2 penggunaan dot)

Fungsi **dot 20 "red"** akan membuat Bidawang menggambar titik dengan ukuran 20 dan warna merah, sedangkan **dot 15 "blue"** akan menggambar titik dengan ukuran 15 dan warna biru.

## Aktivitas 2.8: Menggambar Titik

Yuk kita pelajari bagaimana Bidawang bisa menggambar **titik berwarna** dengan ukuran yang berbeda menggunakan perintah `dot`. Buka aplikasi yang telah disiapkan, kemudian ikuti langkah-langkah di bawah ini secara bertahap. Ketikkan perintahnya satu per satu dan tekan **"Run Code"** untuk melihat hasilnya di canvas.

### Langkah-langkah:

1. Menggambar titik



Buat bidawang menggambar titik dengan warna default dan ukuran 20:

```
dot 20
```

2. Berpindah posisi

Lanjutkan pada baris berikutnya untuk memindahkan posisi bidawang dengan perintah dibawah ini:

```
forward 100
```

3. Menggambar titik

Gambar titik lagi dengan warna biru dan ukuran 20:

```
dot 20 "blue"
```

4. Berpindah posisi

Pindahkan lagi posisi bidawang dengan menggunakan perintah dibawah ini:

```
left 120
```

```
forward 100
```

5. Menggambar titik

Setelah berpindah gambar titik lagi berwarna hijau dengan ukuran 50:

```
dot 50 "green"
```

6. Berpindah posisi

Kemudian pindahkan lagi posisi bidawang ke titik awal dengan perintah home:

```
home
```

## Kesimpulan

Perintah dot berfungsi untuk menggambar titik dengan ukuran dan warna yang dapat disesuaikan. Fungsi ini berguna untuk menandai posisi, membuat pola, atau menambah detail pada gambar yang dibuat.

## Pertanyaan

1. Apa fungsi dari perintah dot?
  - a. Menggambar lingkaran penuh.
  - b. Menggambar busur lingkaran.
  - c. Menggambar titik di posisi saat ini.
  - d. Menghapus titik pada kanvas.
  
2. Apa yang terjadi jika Anda menjalankan `dot 15 "blue"`?
  - a. Titik biru dengan ukuran 15 piksel digambar di posisi saat ini.
  - b. Lingkaran biru dengan ukuran 15 piksel digambar di posisi saat ini.
  - c. Lingkaran penuh biru dengan jari-jari 15 digambar di posisi saat ini.
  - d. Tidak ada yang terjadi.

## A. Position

### Tujuan Pembelajaran

- Memahami fungsi dan penggunaan position().

Fungsi position() digunakan untuk mendapatkan posisi saat ini dari bidawang di dalam canvas. Posisi ini dinyatakan dalam bentuk pasangan koordinat (x, y). Untuk menampilkan koordinatnya kita bisa menggunakan fungsi print().

#### Contoh 1

Menampilkan posisi saat ini dari bidawang:

```
print(position())
```

Hasil:



```
(0.0, 0.0)
```

Gambar 10 (Hasil contoh 1 penggunaan position)

Fungsi position() digunakan untuk mencari tahu posisi bidawang saat ini, kemudian position() akan membuat posisi tersebut di tampilkan pada bagian output log.

#### Contoh 2

Memeriksa posisi Bidawang sebelum dan sesudah pergerakan:

```
# Posisi awal
print("Posisi awal:", position())
# Gerakkan Bidawang ke depan sejauh 100 langkah
forward(100)
# Cek posisi baru
print("Posisi setelah bergerak:", position())
```

Hasil:

```
Posisi awal: (0.0, 0.0)
Posisi setelah bergerak: (100.0, 0.0)
```

*Gambar 11 (Hasil contoh 2 penggunaan position)*

Sebelum di gerakan posisi awal Bidawang adalah (0,0), kemudian setelah digerakan dengan `forward(100)` posisinya menjadi (100, 0).

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `position()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

### 1. Cek Posisi

Lakukan pemeriksaan posisi bidawang terlebih dahulu menggunakan perintah di bawah ini:

```
print(position())
```

### 2. Maju

Kemudian lanjutkan pada baris baru untuk membuat bidawang maju sejauh 150 langkah dengan perintah dibawah ini:

```
forward(150)
```

### 3. Cek Posisi

Lalu periksa lagi posisi bidawang tersebut:

```
print(position())
```

### 4. Pindah Posisi

Selanjutnya pindahkan lagi bidawang ke posisi  $x=75$  dan  $y=75$  dengan perintah di bawah ini:

```
setposition(75,75)
```

#### 5. Cek Posisi

Lanjutkan dengan memeriksa posisi bidawang lagi:

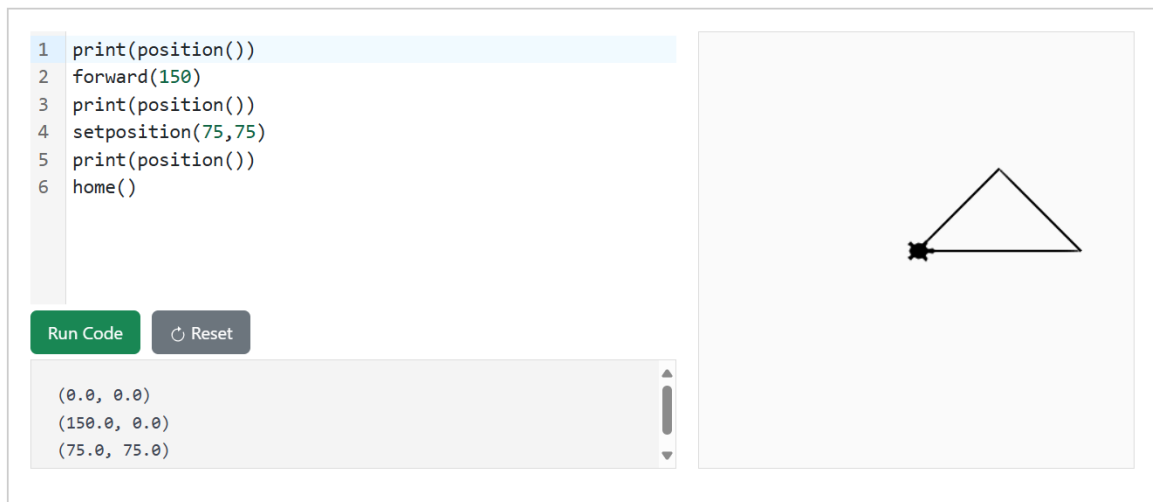
```
print(position())
```

#### 6. Kembali ke posisi awal

Gunakan perintah home untuk membuat bidawang kembali ke titik awal (0,0):

```
home()
```

Jika kalian mengikuti semua instruksi dengan benar, maka hasil akhir pergerakan bidawang akan menggambar segitiga didalam canvas dan akan menampilkan posisi pada bagian output-log seperti dibawah ini:



Gambar 12 (Hasil latihan penggunaan position)

## Kesimpulan

Perintah position() untuk mengetahui posisi saat ini dari Bidawang. Fungsi ini berguna untuk memantau pergerakan dan membantu dalam menggambar pola atau bentuk yang presisi.

## Tantangan

PR

## Kuis

1. Apa fungsi dari `position()`?
  - a. Memberitahu arah pergerakan turtle saat ini.
  - b. Menghapus posisi turtle sebelumnya.
  - c. Memberitahu posisi turtle saat ini dalam bentuk koordinat (x, y).
  - d. Mengatur posisi turtle ke titik tertentu.
  
2. Apa hasil dari perintah berikut jika turtle berada di posisi (50, 100)?  
`print(position())`
  - a. (50, 100)
  - b. 0
  - c. (0, 0)
  - d. [50, 100]

## B. Xcor & Ycor

### Tujuan Pembelajaran

- Mengetahui cara memeriksa posisi horizontal (sumbu x) dan vertikal (sumbu y) dari bidawang.

Fungsi `xcor()` dan `ycor()` digunakan untuk mendapatkan koordinat posisi spesifik di sepanjang sumbu horizontal (x) dan vertikal (y). Metode-metode ini ideal untuk memantau posisi terkini, membantu menentukan apakah bidawang telah mencapai batas tertentu, atau digunakan dalam penghitungan pola geometris yang memerlukan pengawasan koordinat. Sama seperti perintah sebelumnya (*Position()*) Untuk menampilkan hasilnya kita bisa menggunakan fungsi `print()`.

- `xcor()`: Mengembalikan posisi horizontal (sumbu x) bidawang.
- `ycor()`: Mengembalikan posisi vertikal (sumbu y) bidawang.

### Contoh

Menampilkan posisi horizontal (sumbu x) dan vertikal (sumbu y) dari bidawang sebelum dan sesudah bergerak:

```
# Menampilkan posisi awal bidawang
print("Posisi x awal:", xcor())
print("Posisi y awal:", ycor())
# Memindahkan bidawang ke posisi lain
left(45)
forward(150)
# Menampilkan posisi akhir bidawang
print("Posisi x akhir:", xcor())
print("Posisi y akhir:", ycor())
```

Hasil:

```
Posisi x awal: 0.0  
Posisi y awal: 0.0  
Posisi x akhir: 106.0660171779821  
Posisi y akhir: 106.0660171779821
```

Gambar 13 (Hasil contoh penggunaan xcor dan ycor)

Fungsi xcor() akan menampilkan posisi horizontal (sumbu x) dari bidawang, kemudian ycor() akan menampilkan posisi vertikal (sumbu y) dari bidawang.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah xcor() dan ycor() ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

1. Cek posisi x dan y

Lakukan pemeriksaan posisi x dan y menggunakan perintah di bawah ini:

```
print("x : ", xcor())  
print("y : ", ycor())
```

2. Pindahkan posisi

Kemudian lanjutkan lagi pada baris baru untuk menggerakkan bidawang dengan perintah di bawah ini:

```
left(90)  
forward(100)
```

3. Cek posisi x dan y

Lakukan pemeriksaan lagi untuk posisi x dan y menggunakan perintah di bawah ini:

```
print("x setelah bergerak : ", xcor())  
print("y setelah bergerak : ", ycor())
```



## Kesimpulan

Perintah `xcor()` dan `ycor()` berfungsi untuk mengetahui posisi bidawang secara terpisah pada sumbu x dan y. Perintah ini membantu menentukan apakah turtle telah mencapai batas tertentu, atau digunakan dalam penghitungan pola geometris yang memerlukan pengawasan koordinat.

---

## Tantangan

PR

## Kuis

1. Apa perbedaan antara `xcor()` dan `ycor()`?
  - a. `xcor()` menampilkan posisi horizontal (sumbu x), sementara `ycor()` menampilkan posisi vertikal (sumbu y).
  - b. `xcor()` mengatur posisi horizontal, sementara `ycor()` mengembalikan posisi vertikal.
  - c. `xcor()` menghapus posisi pada sumbu x, sementara `ycor()` membaca posisi awal.
  - d. Tidak ada perbedaan, keduanya menampilkan posisi (x, y).
  
2. Apa hasil dari perintah berikut jika turtle berada di koordinat (30, 20)?  
`print(xcor())`
  - a. -20
  - b. 30
  - c. (30, -20)
  - d. 0

## C. Heading

### Tujuan Pembelajaran

- Mengetahui cara memeriksa arah pergerakan dari bidawang menggunakan `heading()`.

Fungsi `heading()` digunakan untuk menampilkan arah pergerakan bidawang dalam satuan derajat. Sama seperti perintah sebelumnya (*Position()*) Untuk menampilkan hasilnya kita bisa menggunakan fungsi `print()`.

### Contoh

Menampilkan posisi arah pergerakan dari bidawang sebelum dan sesudah berotasi:

```
# Periksa arah awal
print("Arah awal:", heading())
# Putar turtle ke kiri sebesar 90 derajat
left(90)
# Periksa arah setelah putaran
print("Arah setelah putaran:", heading())
```

Hasil:

```
Arah awal: 0.0
Arah setelah putaran: 90.0
```

Gambar 14 (Hasil contoh penggunaan `heading`)

Fungsi `heading()` digunakan untuk mendapatkan arah pergerakan dari bidawang, kemudian untuk menampilkan nilainya kita bisa menggunakan fungsi `print()`.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `heading()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `heading()` berguna untuk memantau arah gerakan bidawang dengan lebih tepat. Pemahaman tentang sistem derajat akan membantu siswa menggambar bentuk dan pola secara lebih akurat.

---

## Tantangan

PR

## Kuis

1. Dalam canvas Bidawang, arah 0 derajat mengarah ke mana?
  - a. Utara (atas).
  - b. Timur (kanan).
  - c. Barat (kiri).
  - d. Selatan (bawah).
2. Apa hasil dari perintah berikut jika turtle sudah diputar 90 derajat ke kiri?

```
print(heading())
```

- a. 0
- b. 90
- c. 180
- d. 270

## D. Distance

### Tujuan Pembelajaran

- Memahami cara kerja fungsi `distance()`.

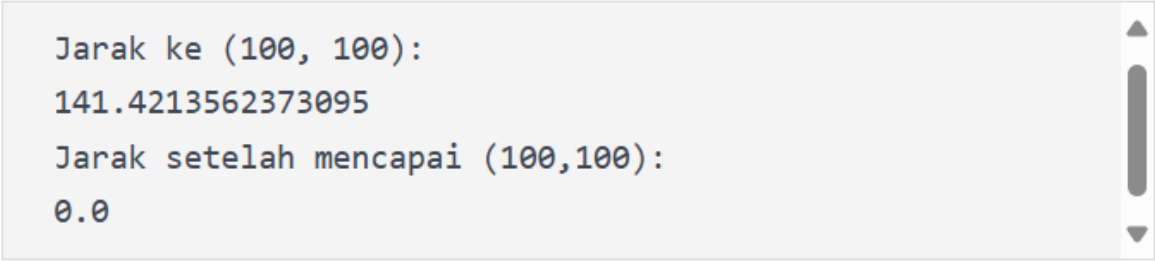
Fungsi `distance()` digunakan untuk menghitung jarak Euclidean antara posisi turtle saat ini dan titik tertentu dengan koordinat yang diberikan. Sama seperti perintah sebelumnya (`Position()`) Untuk menampilkan hasilnya kita bisa menggunakan fungsi `print()`.

### Contoh

Menampilkan jarak posisi bidawang ke koordinat tujuan:

```
# Hitung jarak dari posisi awal ke titik (100, 100)
print("Jarak ke (100, 100):")
print(distance(100,100))
# Gerakkan turtle ke titik (100, 100)
setposition(100, 100)
# Hitung jarak setelah mencapai (100, 100)
print("Jarak setelah mencapai (100,100):")
print(distance(100, 100))
```

Hasil:



```
Jarak ke (100, 100):
141.4213562373095
Jarak setelah mencapai (100,100):
0.0
```

Gambar 15 (Hasil contoh penggunaan `distance`)

Fungsi `distance()` akan mendapatkan jarak posisi bidawang ke koordinat tujuan, kemudian untuk menampilkan nilainya kita bisa menggunakan fungsi `print()`.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `distance()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `heading()` berguna untuk memantau arah gerakan bidawang dengan lebih tepat. Pemahaman tentang sistem derajat akan membantu siswa menggambar bentuk dan pola secara lebih akurat.

---

## Tantangan

PR

## Kuis

1. Apa fungsi dari metode `distance(x, y)`?
  - a. Mengembalikan arah turtle menuju titik  $(x, y)$ .
  - b. Menghitung jarak Euclidean antara posisi saat ini dan koordinat  $(x, y)$ .
  - c. Mengatur turtle untuk bergerak ke posisi  $(x, y)$ .
  - d. Menghapus jarak antara dua titik.

2. Jika turtle berada di koordinat  $(0, 0)$ , apa hasil dari perintah berikut?

```
print(distance(3, 4))
```

- a. 3
- b. 4
- c. 7
- d. 5



## A. Pendown & Penup

### Tujuan Pembelajaran

- Memahami perbedaan antara mode menggambar dan mode tidak menggambar.

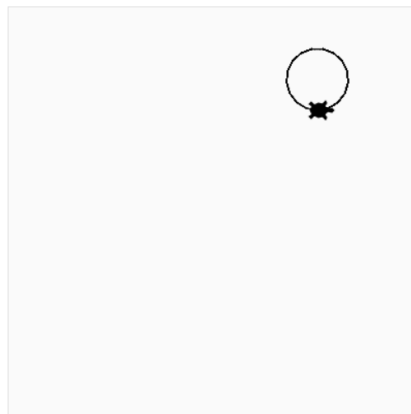
Fungsi `distance()` digunakan untuk menurunkan pena, sehingga turtle akan menggambar garis saat bergerak. Sebaliknya, metode `penup()` mengangkat pena, mencegah turtle menggambar saat bergerak.

### Contoh

Mengaktifkan dan menonaktifkan mode menggambar:

```
# Gerakkan tanpa menggambar
penup()
setposition(100, 100)
# Aktifkan menggambar
pendown()
circle(30)
```

Hasil:



Gambar 16 (Hasil contoh penggunaan `penup` dan `pendown`)

Fungsi `penup()` akan menonaktifkan mode menggambar, jadi meskipun bidawang bergerak tidak akan menggambar garis. Fungsi `pendown()` akan mengaktifkan mode menggambar.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `penup()` dan `pendown()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `pendown()` dan `penup()` sangat penting untuk mengontrol kapan turtle menggambar atau hanya bergerak tanpa meninggalkan jejak. Penggunaan yang tepat membantu dalam pembuatan bentuk yang terpisah.

---

## Tantangan

PR

## Kuis

1. Apa yang terjadi jika metode pendown() tidak dipanggil setelah penup()?
  - a. Bidawang akan terus menggambar saat bergerak.
  - b. Bidawang akan berhenti bergerak.
  - c. Bidawang tidak akan menggambar garis saat bergerak.
  - d. Bidawang akan menggambar lingkaran secara otomatis.

2. Perhatikan kode berikut:

```
penup()  
goto(100, 100)  
pendown()  
43  
goto(200, 200)
```

Apa yang terjadi setelah kode tersebut dijalankan?

- a. Bidawang menggambar garis dari titik awal ke (100, 100).
- b. Bidawang menggambar garis dari (100, 100) ke (200, 200).
- c. Bidawang tidak menggambar sama sekali.
- d. Bidawang hanya menggambar lingkaran.

## B. Pensize

### Tujuan Pembelajaran

- Memahami cara mengubah ketebalan garis menggunakan perintah pensize().

Fungsi pensize(width) digunakan untuk mengatur ketebalan pena saat menggambar garis. Parameter width menunjukkan ukuran ketebalan dalam piksel. Dengan memodifikasi ketebalan garis, pengguna dapat menciptakan variasi visual, seperti membuat garis tebal untuk bingkai atau garis tipis untuk detail halus.

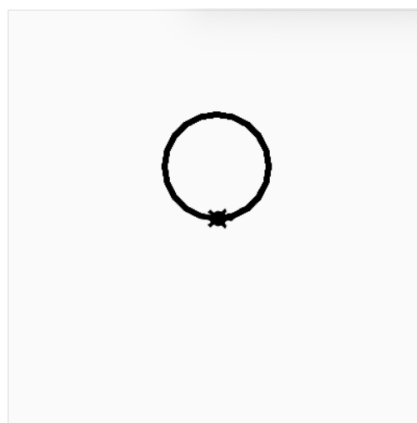
- pensize(width): Mengatur ketebalan garis menjadi width (piksel).

### Contoh

Menggambar lingkaran dengan garis tebal:

```
# Ubah ketebalan pena
pensize(5)
# Gambar lingkaran dengan garis tebal
circle(50)
```

Hasil:



Gambar 17 (Hasil contoh penggunaan pensize)

Fungsi pensize(5) akan membuat ketebalan garis menjadi 5 piksel saat bidawang menggambar lingkaran dengan fungsi circle().

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `pensize()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Dengan perintah `pensize()`, ketebalan garis dapat disesuaikan untuk meningkatkan estetika atau menciptakan variasi visual dalam gambar.

---

## Tantangan

PR

## Kuis

1. Apa efek dari penggunaan metode pensize(10) sebelum menggambar?
  - a. Mengganti warna garis menjadi hitam.
  - b. Mengubah ketebalan garis menjadi 10 piksel.
  - c. Membuat garis tidak terlihat.
  - d. Menghapus garis yang telah digambar.

2. Jika kode berikut dijalankan:

```
pensize(5)  
circle(30)  
pensize(2)  
circle(50)
```

Apa yang akan terlihat pada hasil akhir?

- a. Dua lingkaran dengan ketebalan garis yang sama.
- b. Lingkaran pertama memiliki garis lebih tebal daripada lingkaran kedua.
- c. Lingkaran kedua memiliki garis lebih tebal daripada lingkaran pertama.
- d. Tidak ada lingkaran yang digambar.

## C. Isdown

### Tujuan Pembelajaran

- Memahami cara mengecek status pena apakah sedang aktif menggambar atau tidak.

Fungsi `isdown()` digunakan untuk memeriksa status pena. Perintah ini menyimpan nilai `True` jika pena sedang dalam posisi turun (aktif menggambar) dan `False` jika pena diangkat. Pemeriksaan status pena berguna untuk menentukan apakah Bidawang akan meninggalkan jejak atau tidak saat bergerak.

### Contoh

Mengecek status pena sedang aktif atau tidak:

```
# Cek status pena
print("Status Pena saat ini:")
print(isdown())
# Mengangkat pena
penup()
print("Pena setelah diangkat:")
print(isdown())
```

Hasil:

```
Status Pena saat ini:
True
Pena setelah diangkat:
False
```

Gambar 18 (Hasil contoh penggunaan `isdown`)

Fungsi `isdown()` akan menampilkan `true` jika pena sedang aktif, kemudian ketika pena di angkat dengan fungsi `penup()` maka menampilkan `false`.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `isdown()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `isdown()`, berguna untuk mengecek status pena secara dinamis dan membantu mengontrol logika penggambaran.

---

## Tantangan

PR



## Kuis

1. Apa yang dikembalikan oleh metode `isdown()` jika pena sedang dalam posisi turun?
  - a. True
  - b. False
  - c. None
  - d. Error

2. Perhatikan kode berikut:`pendown()`

```
print(isdown())  
penup()  
print(isdown())
```

Apa output dari kode tersebut?

- a. True, False
- b. False, True
- c. True, True
- d. False, False

## A. Pencolor

### Tujuan Pembelajaran

- Memahami cara mengubah warna garis pada saat menggambar.

Fungsi `pencolor("warna")` fungsi yang digunakan untuk mengontrol warna garis pada gambar yang dihasilkan oleh `bidawang`. Parameter warna bisa berupa nama warna (contoh: "red") atau kode warna RGB (contoh: "FF0000").

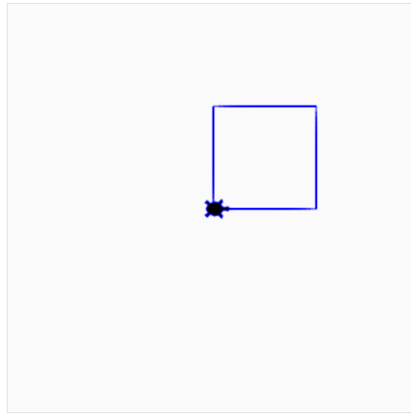
### Contoh

Membuat persegi dengan garis biru:

```
# Mengatur warna garis menjadi biru
pencolor("blue")

# Membuat persegi
forward(100)
left(90)
forward(100)
left(90)
forward(100)
left(90)
forward(100)
left(90)
```

Hasil:



Gambar 19 (Hasil contoh penggunaan pencolor)

Fungsi `pencolor("blue")` akan membuat warna pen jadi biru, jadi ketika bidawang bergerak menggambar persegi maka warna dari garis-garis yang membentuk persegi akan berwarna biru.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `pencolor()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `pencolor()` digunakan untuk mengatur warna garis lintasan yang dilalui oleh bidawang.

## Tantangan

PR

## Kuis

1. Apa yang terjadi jika kode berikut dijalankan?

```
pencolor("red")  
forward(100)
```

- a. Bidawang menggambar garis merah sepanjang 100 piksel.
- b. Bidawang menggambar garis hitam sepanjang 100 piksel.
- c. Bidawang menggambar garis tanpa warna sepanjang 100 piksel.
- d. Bidawang tidak menggambar garis.

2. Apa format yang tidak valid untuk parameter warna dalam fungsi pencolor()?

- a. "blue"
- b. "#00FF00"
- c. (1,0,0)
- d. "bold\_red"

## B. Fillcolor, Begin\_fill, & End\_fill

### Tujuan Pembelajaran

- Memahami cara kerja fungsi fillcolor, begin\_fill, dan end\_fill dalam mengatur warna isian pada bentuk geometris.

Perintah fillcolor("*warna*") Digunakan untuk menentukan warna isian yang akan diterapkan pada bentuk. Parameter warna dapat berupa nama warna atau kode RGB. Untuk mewarnai bentuk, perintah fillcolor("*warna*") harus di sertai dengan begin\_fill() dan end\_fill()

- begin\_fill(): Menandai awal area yang akan diisi warna.
- end\_fill(): Menandai akhir area yang akan diisi warna. Warna isian diterapkan sesuai pengaturan fillcolor.

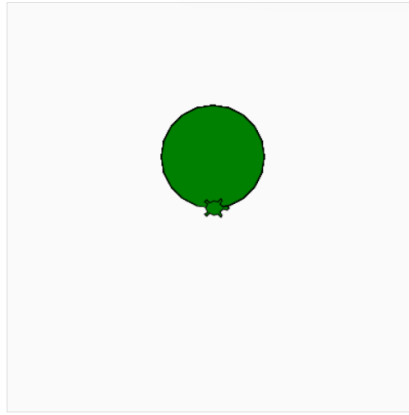
### Contoh

Membuat lingkaran berwarna hijau:

```
fillcolor("green")  # Warna isian hijau

begin_fill()
circle(50)  # Membuat lingkaran
end_fill()
```

Hasil:



Gambar 20 (Hasil contoh penggunaan `fillcolor`, `begin_fill`, dan `end_fill`)

Bidawang akan menggambar lingkaran, kemudian fungsi `fillcolor("green")` akan membuat warna lingkaran tersebut menjadi hijau.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `fillcolor()`, `begin_fill()`, dan `end_fill()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `fillcolor()` digunakan untuk mengatur warna isian, sedangkan proses pengisian warna dilakukan dengan bantuan `begin_fill()` dan `end_fill()`. Untuk hasil optimal, pastikan area yang digambar adalah bentuk tertutup.

## Tantangan

PR

## Kuis

1. Perhatikan kode berikut:

```
forward(100)
begin_fill()
circle(50)
end_fill()
```

Apa hasil dari kode tersebut?

- a. Sebuah lingkaran dengan warna garis kuning tanpa isian.
  - b. Sebuah lingkaran dengan warna isian kuning.
  - c. Sebuah lingkaran dengan warna garis hitam dan isian kuning.
  - d. Tidak ada lingkaran yang digambar.
2. Apa fungsi dari `end_fill()` dalam proses pengisian warna?
- a. Mengatur warna isian menjadi transparan.
  - b. Menandai akhir area yang akan diisi warna.
  - c. Menonaktifkan pengaturan warna pada turtle.
  - d. Menghapus warna isian dari bentuk yang digambar.

## A. Reset

### Tujuan Pembelajaran

- Memahami fungsi reset().

Perintah reset() Digunakan untuk menghapus semua gambar pada canvas dan mengatur ulang posisi serta atribut seperti warna dan ukuran pena, ke nilai default.

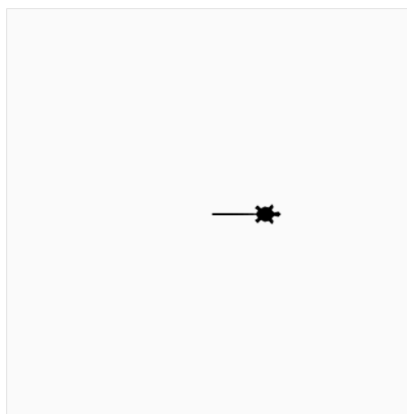
### Contoh

Menghapus gambar yang sudah dibuat dan mengatur ulang atribut serta posisi:

```
pencolor("red")
forward(100)
left(90)
forward(100)

# Reset layar dan posisi turtle
reset()
forward(50) # Memulai gambar baru
```

Hasil:



Gambar 21 (Hasil contoh penggunaan reset)



Bidawang akan menggambar dengan garis merah, kemudian setelah perintah `reset()` semua gambar akan hilang dan posisi serta atribut bidawang akan kembali ke nilai default.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `reset()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `reset()` berguna untuk memulai ulang proses menggambar tanpa harus menghapus semua kode yang sudah ditulis.

---

## Tantangan

PR

## Kuis

1. Apa yang terjadi jika fungsi `reset()` dipanggil setelah menggambar garis?
  - a. Semua gambar dihapus, tetapi atribut bidawang tetap sama.
  - b. Semua gambar dihapus, dan bidawang kembali ke posisi awal dengan atribut default.
  - c. Garis tidak dihapus, tetapi posisi bidawang berubah.
  - d. Bidawang akan keluar dari jendela.

2. Perhatikan kode berikut:

```
pencolor("green")  
forward(100)  
reset()  
circle(50)
```

Apa warna lingkaran yang akan dihasilkan?

- a. Hijau.
- b. Hitam.
- c. Biru.
- d. Merah.

## B. Clear

### Tujuan Pembelajaran

- Memahami fungsi reset().

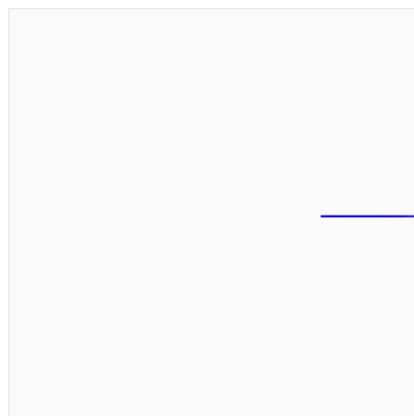
Perintah `clear()` digunakan untuk menghapus semua gambar pada canvas tanpa mengatur ulang posisi atau atribut bidawang.

### Contoh

Menghapus gambar yang sudah dibuat tanpa mengatur ulang atribut dan posisi:

```
pencolor("blue")
forward(50)
circle(50)
# Menghapus gambar tetapi posisi dan atribut tetap sama
clear()
forward(100) # Memulai gambar baru dari posisi terakhir
```

Hasil:



Gambar 22 (Hasil contoh penggunaan `clear`)

Bidawang akan menggambar lingkaran dengan garis biru, Setelah di `clear()` atribut `pencolor "blue"` akan tetap aktif.

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `clear()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `clear()` berguna untuk menghapus gambar tanpa mengubah pengaturan posisi atau atribut bidawang.

---

## Tantangan

PR

## Kuis

1. Apa perbedaan utama antara fungsi `clear()` dan `reset()`?
  - a. `clear()` menghapus gambar tanpa mengubah posisi atau atribut, sedangkan `reset()` juga mengatur ulang posisi dan atribut bidawang.
  - b. `clear()` menghapus gambar beserta posisi bidawang, sedangkan `reset()` hanya menghapus gambar.
  - c. `clear()` tidak menghapus gambar, sedangkan `reset()` menghapus gambar.
  - d. `clear()` menutup jendela, sedangkan `reset()` tidak.

2. Perhatikan kode berikut:

```
color("blue")  
circle(50)  
clear()  
forward(100)  
circle(30)
```

Apa warna lingkaran kedua?

- a. Tidak Berwarna.
- b. Biru.
- c. Hitam.
- d. Merah.

## C. Write

### Tujuan Pembelajaran

- Memahami fungsi write() untuk menampilkan teks pada canvas.

Perintah write() digunakan untuk menulis teks pada canvas di lokasi bidawang berada. write(teks, move=False, align="left", font=("Arial", 8, "normal")):

- Parameter teks: Teks yang akan ditulis.
- Parameter move: Jika True, posisi turtle akan berpindah setelah menulis teks.
- Parameter align: Penyesuaian teks ("left", "center", atau "right").
- Parameter font: Menentukan jenis, ukuran, dan gaya font.

### Contoh

Menulis "Hello, World!" di canvas:

```
write("Hello, World!", align="center", font=("Arial", 16, "bold"))

# Bidawang tetap dapat menggambar setelah menulis
forward(100)
```

Hasil:



Gambar 23 (Hasil contoh penggunaan write)

## Latihan

Untuk lebih memudahkan memahami cara kerja perintah `write()` ikuti instruksi di bawah ini. Kalian bisa membuka aplikasi yang sudah di siapkan untuk menjalankan kodenya, kemudian ketikkan perintah tersebut, lalu tekan tombol "Run Code" untuk melihat pergerakan Bidawang, lakukan secara bertahap:

PR

## Kesimpulan

Perintah `write()` memungkinkan untuk menampilkan teks di layar, sehingga dapat digunakan untuk memberikan informasi tambahan pada gambar.

---

## Tantangan

PR

## Kuis

1. Apa fungsi dari parameter align dalam metode write()?

- a. Mengatur posisi turtle setelah menulis teks.
- b. Menentukan jenis font yang digunakan.
- c. Mengatur perataan teks (kiri, tengah, atau kanan).
- d. Mengatur warna teks.

2. Perhatikan kode berikut:

```
write("Belajar Python!", align="center", font=("Arial",  
12, "italic"))
```

Apa yang akan terjadi?

- a. Teks ditulis di layar dengan font Arial, ukuran 12, dan bergaya italic di posisi kiri bidawang.
- b. Teks ditulis di layar dengan font Arial, ukuran 12, dan bergaya italic di posisi tengah bidawang.
- c. Teks ditulis di layar dengan font Arial, ukuran 12, tetapi tidak bergaya italic.
- d. Tidak ada teks yang ditulis karena font tidak valid.