


Bard Python



Skriv et python script der fungerer som en normal terning. Det skal have en GUI med en knap der kaster "terningen" og kastets resultat nedenunder. 

Se andre kladder 



Her er et Python-script, der fungerer som en normal terning med en GUI:

Python

```
import tkinter as tk

def kast():
    # Kaster terningen
    kast = randint(1, 6)

    # Opdaterer resultatet i GUI'en
    resultat_label.config(text=f"Resultat: {kast}")

# Opretter GUI'en
root = tk.Tk()
root.title("Terning")

# Skaber en knap
kast_knap = tk.Button(root, text="Kast", command=kast)
kast_knap.pack()

# Skaber en label til at vise resultatet
resultat_label = tk.Label(root, text="Resultat:")
resultat_label.pack()

# Kører GUI'en
root.mainloop()
```

Brug koden med omtanke. [Få flere oplysninger](#)



```
resultat_label = tk.Label(root, text="Resultat:")
resultat_label.pack()

# Kører GUI'en
root.mainloop()
```

Brug koden med omtanke. [Få flere oplysninger](#)



Dette script bruger `randint()`-funktionen fra `random`-modulet til at generere et tilfældigt tal mellem 1 og 6. Tallet gemmes i en variabel kaldet `kast`.

Når brugeren klikker på knappen, kaldes `kast()`-funktionen, der opdaterer resultatet i GUI'en.

GUI'en består af to elementer: en knap og en label. Knappen har teksten "Kast", og når den klikkes, kaldes `kast()`-funktionen. Labelen har teksten "Resultat:", og den opdateres med resultatet af terningskastet.

Her er et eksempel på, hvordan GUI'en ser ud:

```
+-----+
| Terning |
+-----+
| Kast    |
+-----+
| Resultat: |
+-----+
```

Du kan selvfølgelig tilpasse scriptet efter dine behov. For eksempel kan du tilføje flere terninger, ændre antallet af sider på terningen eller tilføje funktionalitet til at gemme resultaterne af kast.



Skriv en prompt her





Terning



Kast

Resultat:

Bard.py

```
1 import tkinter as tk
2 import random
3
4 def kast():
5     # Kaster terningen
6     kast = randint(1, 6)
7
8     # Opdaterer resultatet i GUI'en
9     resultat_label.config(text=f"Resultat: {kast}")
10
11 # Opretter GUI'en
12 root = tk.Tk()
13 root.title("Terning")
14
15 # Skaber en knap
16 kast_knap = tk.Button(root, text="Kast", command=kast)
17 kast_knap.pack()
18
19 # Skaber en label til at vise resultatet
20 resultat_label = tk.Label(root, text="Resultat:")
21 resultat_label.pack()
22
23 # Kører GUI'en
24 root.mainloop()
25
```