

Badger
2040 

PIMORONI 

Dino Game

Press UP to start, A to abort

High score: 3

Press B to upload score



DinoGame

@JessTemporal

jtemporal.com

```
from auth import device_flow as df
from auth.auth0 import config_auth0
from dino.leaderboard import set_user_score

# . . . previous code
elif display.pressed(BUTTON_B):
    print("Connecting to the internet...")
    if not display.isconnected():
        display.connect()
    clear_screen()

    print("Config Auth0")
    auth0_request_data = config_auth0()

    print("Requesting tokens")
    tokens = df.login(auth0_request_data)

    print("Setting user score with API...")
    set_user_score(tokens.get('access_token', 'error'), score)
    start_text()
```







Dino Game

```
from auth import device_flow as df
from auth.auth0 import config_auth0
from dino.leaderboard import set_user_score

# . . . previous code
elif display.pressed(BUTTON_B):
    print("Connecting to the internet...")
    if not display.isconnected():
        display.connect()
    clear_screen()

    print("Config Auth0")
    auth0_request_data = config_auth0()

    print("Requesting tokens")
    tokens = df.login(auth0_request_data)

    print("Setting user score with API...")
    set_user_score(tokens.get('access_token', 'error'), score)
    start_text()
```



Device Flow

```
def config_auth0():
    auth0_config = open(AUTH0_CONFIG_PATH, "r")

    AUTH0_DOMAIN = auth0_config.readline()
    AUTH0_CLIENT_ID = auth0_config.readline()
    ALGORITHMS = auth0_config.readline()
    AUDIENCE = auth0_config.readline()

    auth0_request_data = {
        "domain": AUTH0_DOMAIN.strip('\n'),
        "client_id": AUTH0_CLIENT_ID.strip('\n'),
        "algorithms": [ALGORITHMS.strip('\n')],
        "audience": AUDIENCE.strip('\n')
    }

    auth0_config.close()
    return auth0_request_data
```



```
from auth import device_flow as df
from auth.auth0 import config_auth0
from dino.leaderboard import set_user_score

# . . . previous code
elif display.pressed(BUTTON_B):
    print("Connecting to the internet...")
    if not display.isconnected():
        display.connect()
    clear_screen()

    print("Config Auth0")
    auth0_request_data = config_auth0()

    print("Requesting tokens")
    tokens = df.login(auth0_request_data)

    print("Setting user score with API...")
    set_user_score(tokens.get('access_token'))
```