

Begin Modern Programming

with

C
O
D
I
T
Y
A

Pyi Soe

အခန်း ၁

စက်ရုပ်ကားရဲလိုဖြင့် ပရီဂရမ်းမင်းမိတ်ဆက်

ကွန်ပျူတာတွေဟာ သက်မဲ့ စက်ပစ္စည်းတွေပါပဲ။ ကားတို့ လေယာဉ်တို့နဲ့ မတူတာက ကွန်ပျူတာတွေဟာ စက်ချဉ်းသက်သက် ဘာအစွမ်းမှ မယ်မယ်ရရ မရှိဘူး။ ဒါပေမဲ့ ဆောင်ရွက်လိုတဲ့ ကိစ္စအဝေဝအတွက် ပရီဂရမ်းမျိုးမျိုး ထည့်ပေးလိုက်တဲ့ အခါမာ သူ့အစွမ်းက အတိုင်းအဆမဲ့ပဲ။ နေရာမျိုးစံ၊ နယ်ပါယ်မျိုးစံ မှာ အကူးအညီပေးနိုင်တဲ့ စွယ်စုံသုံး ပစ္စည်းတစ်ခုဖြစ်သွားတယ်။ ဂိတ်သံစဉ်တွေကို ဖွင့်ပေးနိုင်သလို အသံလည်းသွင်းပေးနိုင်တယ်။ ရုပ်ရှင်တည်းဖြတ် လုပ်ချင်တာလား။ ပြဿနာမရှိဘူး၊ ကူညီပေးနိုင်တယ်။ နျှော လီးယား ပါတ်ပေါင်းဖို့တွေကို စီမံခိုင်သလို မောင်းသူမဲ့ အုံပုံတွေကိုလည်း ပဲထိန်းပေးနိုင်တယ်။

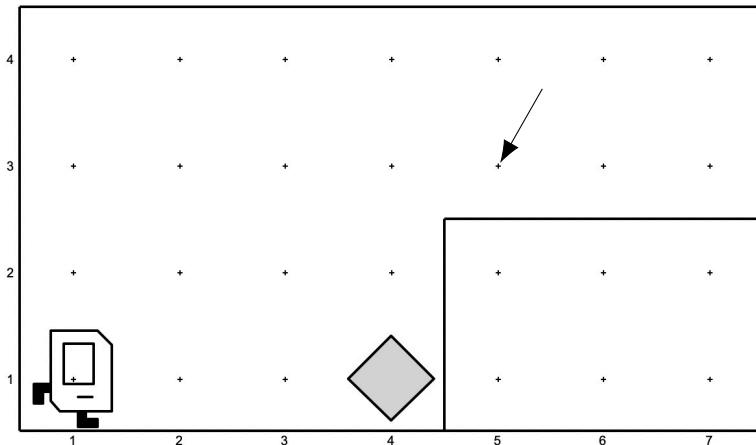
ကျွန်တော်တို့တွေ နိစ္စဓာဝ အသုံးပြုနေကြတဲ့ ကား၊ စမတ်နှင့်ဗုံး လက်ပါတ်နာရီ၊ မိုက်ခရှိဝေဖွံ့ဖို့ အဝတ်လျှော့စက် စတဲ့ စက်ပစ္စည်း အမျိုးမျိုးဟာလည်း ကွန်ပျူတာတွေနဲ့ မကင်းပြန်ပါဘူး။ “ကွန်ပျူတာနည်းပညာ အကူးအညီမပါတဲ့ အတိုင်းဆက်သံတိထွင်မှုဆိုတာ မရှိဘူး” လို့ ဆိုနိုင်ပါတယ်။

တစ်ချက်တစ်ချက် ရိုက်ခတ်လိုက်တဲ့ ကွန်ပျူတာနည်းပညာ လိုင်းလုံးကြီးတွေဟာ ကမ္မာတစ်စုံမှုးလုံး ပုံစံပြောင်းသွားလောက်အောင် အဟုန်ပြင်းထန်လာတယ်။ ဘီလီယံချွဲခြုံတဲ့ လူတွေ ဆီရှယ်မီဒီယာတွေပေါ်က နေ ရုပ်သံတွေနဲ့ ချိတ်ဆက်ပြောဆိုသွေးတွေ ရစေတာဟာလည်း ကွန်ပျူတာစနစ်တွေပါပဲ။ Artificial Intelligence (AI) နည်းပညာကြောင့် သက်ရှိတွေမှာပဲတွေ့ရတဲ့ ညာတ်ရည်မျိုးကို ကွန်ပျူတာတွေမှာလည်း တွေ့လာရပါပြီ။ သံချွဲပွဲတွေ ဖြေရှင်းခြင်း၊ စစ်တုံးရှင်းထိုးခြင်း၊ စတဲ့ကိစ္စမျိုးတွေအပြင် ပန်းချီဆွဲခြင်း၊ ကျွေးမှုရေးစပ်ခြင်း၊ သီချင်းရေးဖွဲ့ခြင်း ကဲ့သို့ အနုပညာဖန်တီးမှုတွေကိုပါ AI က လုပ်ဆောင်ပေးနိုင်ပါတယ်။ နှစ်ဆယ်တစ်ရာစုံ၊ အထူးဤားဆုံး AI နည်းပညာလိုင်းဟာ အရှိန်အဟုန်ပြင်းပြင်း ရိုတ်ခတ်ဖို့ အားယူစ ပြုနေပါပြီ။

‘ကွန်ပျူတာ’ လိုပြောတဲ့ အခါ စက်ပစ္စည်းသက်သက် မဟုတ်ဘဲ ကွန်ပျူတာမှုတ်ညာတ်တဲ့ ပရီဂရမ်တွေလည်း ပါဝင်တယ်ဆိုတာ သတိချုပ်ရပါမယ်။ ကွန်ပျူတာတွေ တစ်စုံတစ်ရာ စွမ်းဆောင်နိုင်စေတဲ့ ပရီဂရမ်တွေ ရေးတဲ့ အလုပ်ကို ပရီဂရမ်းမင်း(Programming) လို့ ခေါ်တယ်။

၁.၁ စက်ရုပ် ကားရဲလို

ပရီဂရမ်းမင်းဆိုတာ ဘယ်လိုမျိုးလဲ သဘောပေါက်အောင် စာတွေတစ်သီဥပြီးရေး ရှင်းပြတာထက် ပရီဂရမ်လေးတွေ လက်တွေ၊ ရေးကြည့်လိုက်တာ ပိုပြီးထိရောက်ပါတယ်။ ဒါကြောင့် စက်ရုပ်ကားရဲလိုကို ပရီဂရမ်လေးတွေရေးပြီး အလုပ်တွေခုံးကြည့်ကြမယ်။ ပုံ (၁.၁) မှာ တွေ့ရတာက ကားရဲလို ရောက်ရှိနေတဲ့ နူးနာ ကမ္မာတစ်ခုပါ။ မီးခုံးရောင် မှန်ကူကူပုံလေးကို ဘိပါ (beeper) လို့ ခေါ်တယ်။ အဲဒီဘိပါကို မြားပြထားတဲ့ နေရာကို ရွှေခိုင်းချင်တယ်။ မျှေားမည်းအထူးတွေက နံရံတွေပါ။ ကားရဲလိုကို ကိစ္စတစ်ခု ဆောင်ရွက်စေ



ပုံ ၁.၁ ခက်ရုပ်လေး ကားရဲလ်

ချင်တဲ့အခါ အခြေခံ ကားရဲလ်ကွန်မန်းတွေကို အသုံးပြုရပါတယ်။ ကွန်မန်းတွေကို နှုတ်နှုံးပြောပြီး ခိုင်းရတာ မဟုတ်ဘဲ ပရိုကရမ်းပြီး ခိုင်းရတာပါ။ ကားရဲလ်နားလည်တဲ့ ကွန်မန်းတွေကို ကြည့်ကြရအောင်။

ကားရဲလ်ကွန်မန်းများ

မဖြစ်မနေ သိထားရမဲ့ အခြေခံ ကားရဲလ်ကွန်မန်း လေးခုပဲ ရှိတယ်။ move, turn_left, put_beeper နဲ့ pick_beeper တို့ဖြစ်တယ်။ အခြား ကားရဲလ်ကွန်မန်း တွေလည်း ရှိပါသေးတယ်။ ဒါပေမဲ့ ကားရဲလ် ပရိုကရမ်းမင်း စလေ့လာဖို့ ဒီလေးခုနဲ့ပဲ လုံလောက်ပါပြီ။

move ကွန်မန်းက ကားရဲလ်ကို ရှေ့တစ်ကွန်နာကို ရွှေ့ခိုင်းတာ။ ကားရဲလ်ကမ္ဘာထဲမှာ တစ်ခုနဲ့တစ်ခု အကွာအဝေးတူ ခြားထားတဲ့ အတန်းလိုက် အတန်းလိုက် အစက်ကလေးတွေဟာ ကွန်နာ (corner) တွေ ဖြစ်တယ်။ ကမ္ဘာကို ပျော်မည်းမည် အထူး နံရုံတွေနဲ့ ထောင့်မှန်စတုဂံပဲ ပါတ်လည် ဘောင်ခတ်ထားတယ်။ ကွန်နာတွေကြားမှာလည်း နံရုံတွေရှိနိုင်တယ်။ နူးနာကမ္ဘာမှာ ဘေးတိုက် နံပါတ်စဉ် ၄ နဲ့ ၅ ကြား ထောင်လိုက် နံရုံတစ်ခု၊ အထက်အောက် နံပါတ်စဉ် ၂ နဲ့ ၃ ကြား အလျေားလိုက် နံရုံတစ်ခုကို တွေ့ရှုပါမယ်။ ကွန်နာရှေ့မှာ နံရုံကာနေရင် ကားရဲလ်ကို move ခိုင်းလို့မရပါဘူး။

put_beeper က ကားရဲလ် လက်ရှိ ရှိနေတဲ့ ကွန်နာမှာ ‘ဘိပါတစ်ခုချု’ ထားခိုင်းတာ၊ pick_beeper က ရပ်နေတဲ့ ကွန်နာမှာ ‘ဘိပါတစ်ခုကောက်’ ခိုင်းတာပါ။ ကွန်နာမှာ ဘိပါရှိနေမှ ကောက်ခိုင်းလို့ရမှုပါ။ မရှုရင် ကောက်ခိုင်းလို့ မရှုဘူး။ ဘိပါချိခိုင်းရင်လည်း ကားရဲလ်မှာ ဘိပါရှိမှ ချိခိုင်းလို့ရတယ်။ ကားရဲလ်ကို ဘိပါတွေ လိုသလောက် ဖြည့်ပေးထားတယ်လို့ ယူဆပါ။ turn_left က ‘ဘယ်လှည့်’ ခိုင်းတာ။

ဘိပါကို ဘယ်လိုရွှေ့ခိုင်းမလဲ

ပုံ (၁.၁) အနေအထားကနေ ရေ့ကို သုံးနေရာရွှေ့ ဘိပါကောက်၊ ဘယ်ဘက်လှည့်၊ အပေါ် နှစ်နေရာရွှေ့ ညာဘက်လှည့်၊ ရှေ့တစ်နေရာထပ်ရွှေ့ပြီး ဘိပါချုထားခိုင်းလိုက်ရင် အလုပ်ပြီးသွားပါပြီ။

ကားရဲလ်ကို ညာဘက်လှည့်ခိုင်းဖို့ turn_right ကွန်မန်း မရှိဘူး။ ဒါပေမဲ့ ဘယ်သုံးခါလှည့်တာဟာ ညာဘက်လှည့်တာနဲ့ တူတူပါပဲ။ ဒါကြောင့် ညာဘက်ချင်တဲ့အခါ ဘယ်သုံးခါလှည့်ခိုင်းလို့ရတယ်။

၁.J Meet Karel ပရိုဂရမ်

ပရိုဂရမ် ရေးတယ်ဆိုတာ ကွန်ပျူးတာကို ကိစ္စတစ်ခုခဲ့ အောက်ရှုက်ပေးဖို့ ခိုင်းစေတဲ့ ညွှန်ကြားချက်တွေ ရေးတာပါပဲ။ ဒီလို ညွှန်ကြားချက်တွေကို ပရိုဂရမ်ကုဒ် (program code) လို ခေါ်တယ်။ ပရိုဂရမ်ကုဒ် တွေကို ကွန်ပျူးတာနားလည်တဲ့ programming language တစ်ခုခဲ့ ရေးရတယ်။ ဒီစာအုပ်မှာ အသံးပြုမဲ့ programming language ကတော့ Python ပါ။ Programming language တစ်မျိုးပဲ ရှိတာ မဟုတ်ပါဘူး။ ရာနဲ့ချိပြီး ရှိတာပါ။ လူဘာသာစကားတွေ အမျိုးမျိုးရှိသလိုပဲပေါ့။ Python ဟာ ဒီလို ရာနဲ့ချိတဲ့ထဲက လက်ရှိအသံးအများဆုံး ထိပ်ဆုံးဆယ်ခု ထဲမှာ ပါဝင်တယ်။ Python နဲ့ ဘိပါန္တာခိုင်းတဲ့ ပရိုဂရမ်ကို လေ့လာကြည့်ရအောင်။ ကားရဲလ်နဲ့ ပထမဆုံး မိတ်ဆက်ပေးတဲ့ ပရိုဂရမ်မိုလို ဒီပရိုဂရမ် နံမည်ကို ‘Meet Karel’ လို ခေါ်ပါမယ်။

```
# File: meet_karel.py
# About: This is
from stanfordkarel import *

def main():
    """Karel code goes here!"""
    move()
    move()
    move()
    pick_beeper()
    turn_left()
    move()
    move()

    turn_left()
    turn_left()
    turn_left()

    move()
    put_beeper()
# End of main

if __name__ == "__main__":
    run_karel_program("meet_karel")
```

ဒါဟာ ‘Meet Karel’ ပရိုဂရမ်အတွက် Python နဲ့ရေးထားတဲ့ ပရိုဂရမ်ကုဒ် တွေဖြစ်ပါတယ်။ ‘Python ကုဒ်’ လို အတိုကျိုးပဲ ပြောတာများတယ်။ Python ‘စာ/စကား’ မတတ်ရင် ဒါ ‘Python ကုဒ်’ တွေကိုလည်း နားလည်မှာ မဟုတ်ဘူး။ ဒီတော့ Python ‘စာ/စကား’ အခြေခံက စပြီး လေ့လာဖို့လိုပါမယ်။

ကွန်းမာန် (Comment)

ပထမဆုံး # သင်္ကာနဲ့ စတဲ့ စာကြောင်းတွေက ကွန်းမာန်တွေပါ။ ကွန်းမာန်တွေက ကွန်ပျူးတာ အောင်ရှုက်ပေးရမဲ့ ညွှန်ကြားချက်တွေ မဟုတ်ဘူး။ ပရိုဂရမ်ကုဒ်နဲ့ ပါတ်သက်ပြီး ကုဒ် ဖတ်ရှုသူ အတွက် မှတ်ချက်ရေးတာ သို့မဟုတ် ရှင်းပြထားတာပါ။ တနည်းအားဖြင့် ဖတ်ရှုသူ (လူ) ပရိုဂရမ်မှာအတွက် ရည်ရွယ်တာ။ ကွန်ပျူးတာ (စက်) အတွက် ရည်ရွယ်တာ မဟုတ်ဘူး။ ကွန်ပျူးတာက ကုဒ်တဲ့က ကွန်းမာန်တွေ အားလုံးကို

လစ်လၢြရမှာပါ။ ဒါပေမဲ့ ပရိုကရမ်ကုဒ်ကို ဖတ်တဲ့လူ နားလည်ဖို့ အထောက်အကူးဖြစ်စေတဲ့အတွက် ကုန်းမန်ရေးတာကို ပေါ့ပေါ့တန်တန် အရေးမပါသလို သဘောထားလို့ မရပါဘူး။ မိမိရေးတဲ့ ကုဒ်ကို ရှင်းပြနိုင် လိုအပ်ရင် ကုန်းမန်ရေးရပါမယ်။ ရေးသင့်တဲ့ နေရာတွေကိုလည်း မကြာခင်တွေရမှာပါ။

import စတိတ်မန္တ

```
from stanfordkarel import *
```

ကတော့ အင်ပို့စတိတ်မန် ဖြစ်ပါတယ်။ “stanfordkarel လိုက်ဘရီမှ အာလုံးကို ထည့်သင်းပေးပါ” လိုတောင်းဆိုတဲ့ အဓိပ္ပာယ်။ * သက်ဗောက်ကို ‘အားလုံး’ လို့ ယူဆပါ။ stanfordkarel လိုက်ဘရီမှာ ကားရဲ့လုပ်ပရိုကရမ်အတွက် လိုအပ်တာအားလုံး ပါဝင်တယ်။ ဒီလိုက်ဘရီကို အင်ပို့လုပ်ထားမှ ကားရဲ့လုပ်ကွန်းများတွေ သုံးလို့ရမှာပါ။ သီးခြား ကားရဲ့လုပ်ပရိုကရမ် တစ်ခုစီတိုင်းအတွက် အင်ပို့လုပ်ရမှာ ဖြစ်တယ်။

လိုက်ဘရီ

လိုက်ဘရီ (*library*) ဆိုတာ ပညာရပ်နယ်ပယ် တစ်ခုအတွက် ရည်ရွယ်ရေးထားတဲ့ ပရိုကရမ်ကုဒ်တွေပါပဲ။ သချုပ်အတွက်အချက် လိုက်ဘရီ ဂိမ်းရေးစွဲ လိုက်ဘရီ၊ 2D/3D ရေပ်ဖွဲ့စွဲ လိုက်ဘရီ၊ အေအာင်အတွက် လိုက်ဘရီ စသည်ဖြင့် နယ်ပယ်အသီးသီး၊ ကိစ္စရပ်အဖို့ဖို့အတွက် သက်ဆိုင်ရာ ကျမ်းကျင်ပညာရှင်တွေ ထုတ်လုပ်ဖြန့်ချီပေးထားတဲ့ လိုက်ဘရီတွေရှိတယ်။ Matrix အော်ပရေးရှင်းတွေအတွက် numpy၊ ဂရပ်စွဲမယ်ဆိုရင် matplotlib စတဲ့ လိုက်ဘရီတွေကို အင်ပို့လုပ် အသုံးပြနိုင်ပါတယ်။ မူထရာစ် A ကို B နဲ့ မြောက်ရင် ဒီလိုပါ

```
from numpy import *
```

```
A = array([[1, 1, 2, 2],
           [2, 2, 1, 1],
           [2, 2, 1, 1]])
B = array([[2, 2],
           [2, 2],
           [1, 1],
           [2, 2]])

result = matmul(A, B)

print(result)
```

ရလဒ် အခုလိုထွေက်ပါမယ်။

```
[[10 10]
 [11 11]
 [11 11]]
```

ဒါကတော့ ဘားချုပ် အတွက် numpy နဲ့ matplotlib လိုက်ဘရီ သုံးထားတာပါ။

```
from matplotlib.pyplot import *
from numpy import *
```

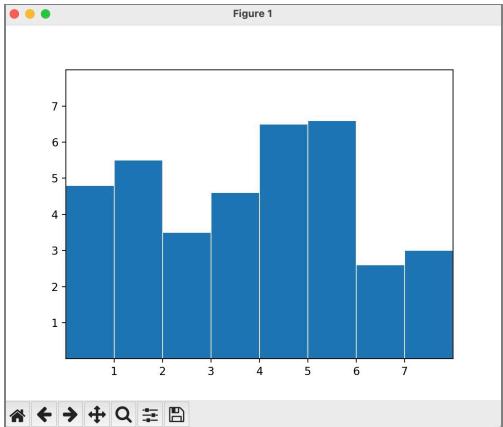
```
# make data:
x = 0.5 + arange(8)
y = [4.8, 5.5, 3.5, 4.6, 6.5, 6.6, 2.6, 3.0]
```

```
# plot
fig, ax = subplots()
bar(x, y, width=1, edgecolor="white", linewidth=0.7)

ax.set(xlim=(0, 8), xticks=arange(1, 8),
       ylim=(0, 8), yticks=arange(1, 8))

show()
```

ဘားချုတ်ကို ဒီလို ထုတ်ပေးပါတယ်။



ထဲ ၁၂

လိုက်ဘရီတွေဟာ ပရိုဂရမ်တွေ တည်ဆောက်ရာမှာ အင်မတန်မှ အရေးပါတယ်။ ဖော်ပြထားတဲ့ မေ့သရစ် နဲ့ ဘားချုတ် ကုဒ်တွေကို (အခုတော့) နားလည်မှာ မဟုတ်သေးဘူးပေါ့။ ဒါပေမဲ့ သက်ဆိုင်ရာ လိုက်ဘရီတွေနဲ့ ဒီလိုကိစ္စတွေကို သိပ်မခက်ခဲဘဲ လုပ်လိုရနိုင်တယ်ဆိုတာ မြင်မယ် ထင်ပါတယ်။ လိုက်ဘရီတွေ သာမရှိရင် ပရိုဂရမ်တွေကို အခုထက် အဆပေါင်းများစွာ အချိန်ပေးပြီး ရှုပ်ရှုပ်ထွေးထွေး ခက်ခက်ခဲ့လဲ တည်ဆောက်ကြရမှာပါ။

တံ့ကင်၊ စတိတ်မန်နှင့် ဆင်းတက်ခြင်း

ဂျို့နှင့် ပြင်သစ်စ စတဲ့ လူဘာသာစကားတွေဟာ စကားလုံးတွေ ဝါကျတွေနဲ့ ဖွဲ့စည်းထားသလို ပရီ ဂရမ်ကုဒ်တွေဟာလည်း စကားလုံးတွေ ဝါကျတွေနဲ့ ဖွဲ့စည်းထားတာပါပဲ။ Programming language တွေမှာ စကားလုံးတွေကို တံ့ကင် (token) လိုခေါ်ပြီး ဝါကျတွေကိုတော့ စတိတ်မန် (statement) လို ခေါ်ပါတယ်။ ဝါကျတွေကို စကားလုံးတွေနဲ့ ဖွဲ့စည်းထားသလို စတိတ်မန်တွေကတော့ တံ့ကင်တွေနဲ့ ဖွဲ့စည်းထားတာပါ။ စတိတ်မန် ပုံစံတစ်မျိုးကို တွေ့ခဲ့ပြီးပါပြီ။ အဲဒါကတော့ ရေ့စာမျက်နှာက အင်ပိုစတိတ်မန်ပဲ ဖြစ်ပါတယ်။

လူဘာသာစကားတွေမှာ ဖွဲ့စည်းတည်ဆောက်ပုံ စထရက်ချာရှိသလို programming language တွေမှာလည်း စထရက်ချာရှိဖို့ လိုအပ်တာပေါ့။ ဖွဲ့စည်းပုံ စထရက်ချာ မှန်/မမှန်ကို သဒ္ဓါစည်းမျဉ်တွေနဲ့ ထိန်းကွာ်ထားတာပါ။ ပရိုဂရမ်ကုဒ် စထရက်ချာ မှန်/မမှန် ထိန်းကွာ်ပေးတဲ့ သဒ္ဓါစည်းမျဉ်တွေကိုတော့ ဆင်းတက်ခြင်း (syntax) လိုခေါ်တယ်။

မြန်မာလိုရေးရင် မြန်မာသဒ္ဓါကို လိုက်နာရသလို Python နဲ့ ရေးရင် Python ဆင်းတက်စိုးကို

လိုက်နာရမှာပေါ့။ မြန်မာသွေ့မှားရင် ဖတ်တဲ့သူက သည်းခံနားလည် ပေးပေမဲ့ ဆင်းတက်စုံမှားရင်တော့ Python က လုံးဝ လက်ခံမှာ မဟုတ်ပါဘူး။ ဆင်းတက်စုံ စဉ်းကမ်းတွေဟာ ပိုပြီး တင်းကျပ်တယ်။ လွှဲချော်လို့ မရဘူး။ ဆင်းတက်စုံမှားနေတဲ့ ပရိုဂရမ်ကို Python က run ခွင့် ပြုမှာမဟုတ်ဘဲ အမှားနဲ့ သက်ဆိုင်တဲ့ အယ်ရာမက်ဆွဲချုပ်တွေ ပြပေးမှာပါ။ ဖြစ်လေရှိတဲ့ ဆင်းတက်စုံအမှားတွေကို မကြာခင် တွေ ရပါမယ်။

Keywords

`from, import, def, if` စတာတွေဟာ *keyword* တွေဖြစ်တယ်။ Python ရေးတဲ့အခါ သူနေရာနဲ့ သူ အဓိပ္ပာယ်ကိုယ်စိန့် အသုံးပြုရတဲ့ စကားလုံးတွေဖြစ်တယ်။ `from` နဲ့ `import` ကို လိုက်ဘရီ အင်ပိုလုပ် ဖို့ သုံးတယ်။ `def` ကို ဖန်ရှင် သတ်မှတ်တဲ့အခါ သုံးတယ်။ Python က သတ်မှတ်ထားတဲ့ နေရာတွေက လွှဲလို့ အခြားကိစ္စတေအတာက *keyword* တွေကို အသုံးပြုလို့ မရပါဘူး။ ဒါကြောင့် *keyword* တွေကို reserved word လို့လည်း ခေါ်တယ်။

main ဖန်ရှင်

‘Meet Karel’ ပရိုဂရမ် အင်ပိုစတိတ်မန့် အပြီးမှာ တွေ့ရတာကတော့ `main` ဖန်ရှင်သတ်မှတ်ချက်ပါ။ ကြည့်ရအဆင်ပြေအောင် သူချည်းသီးသန့် တစ်ဖြတ် ပြန်ပြပေးထားပါတယ်။

```
def main():
    """Karel code goes here!"""
    move()
    move()
    move()
    pick_beeper()
    turn_left()
    move()
    move()

    turn_left()
    turn_left()
    turn_left()

    move()
    put_beeper()
# End of main
```

ဖန်ရှင် (*function*) ဆိုတာ ကိစ္စတေစုံ လုပ်ဆောင်ပေးဖို့အတွက် ယူနစ်တစ်ခုအနေနဲ့ ဖွဲ့စည်းထားတဲ့ ပရိုဂရမ်ကုဒ် အစုအဝေးတစ်ခုပါပဲ။ ဖန်ရှင်ကို အသုံးပြုတဲ့အခါ ငှါးရဲ့ လုပ်ငန်းတာဝန်အတိုင်း ဖန်ရှင်က လုပ်ဆောင်ပေးမှာ ဖြစ်တယ်။ ဖန်ရှင်အသုံးပြုတာကို ‘ဖန်ရှင်ကောလ်’ (*function call*) လုပ်တာလို့ ပြောတယ်။

`main` ဖန်ရှင်သတ်မှတ်ချက်ကို အပိုင်းနှစ်ပိုင်းခဲ့ ကြည့်နိုင်တယ်။ ပထမတစ်ပိုင်း

```
def main():
```

ကို ဖန်ရှင်ခေါင်းစီး (function header) လို့ခေါ်တယ်။ ဖန်ရှင်ခေါင်းစီးမှာ ဖန်ရှင်နံမည်နဲ့ ဖန်ရှင်ပါရာ မိတာတွေကို ပိုက်ကွင်းထဲမှာ သတ်မှတ်ပေးရပြီး ကော်လုံးနဲ့ အဆုံးသတ်တယ်။ ဥပမာ x, y ပါရာ

မိတ္တ နဲ့ myfun ဖန်ရှင် အတွက်

| **def myfun(x, y):**

ပါရာမီတာမပါရင်လည်း ပိုက်ကွင်းအလွှတ် တစ်စုံ () တော့ပါရမယ်။ main ဖန်ရှင်မှာ ပါရာမီတာ မပါဘူး။ ပါရာမီတာတွေအကြောင်း နောက်ပိုင်းအခန်းတွေမှာ အသေးစိတ် လေ့လာရမှာပါ။ ကားချုပ်ပရှိရမ်တွေမှာ ပါရာမီတာအကြောင်း သိမ့်မလိုသေးပါဘူး။ ပါရာမီတာ မလိုတဲ့ ဖန်ရှင်တွေပဲ တွေ့ရမှာပါ။

ဖန်ရှင်သတ်မှတ်ချက် ဒုတိယပိုင်းကတော့ ဖန်ရှင်သော်ဒါပါ။ ဖန်ရှင်သော်ဒါ ရေးရင် ဘေးမျဉ်းကနေ ညာဘက်ကို ဆွဲရေးရပါမယ်။ ခွာတဲ့ အက္ဂာအဝေး တာပြေးညီးဖြစ်ရမယ်။ အင်ဒန်ထုတ် (indent) လုပ်တာ လို့ ခေါ်တယ်။ ကုပ်စထရှင်ချာကို ကြည့်လိုက်တာနဲ့ ထင်းကနဲ့ မြင်သာအောင် လုပ်ရတာပါ။ main ဖန်ရှင်မှာ ခေါင်းစီးအောက် အင်ဒန်ထုတ်လုပ်ထားတဲ့ ကုပ်အားလုံးဟာ ဖန်ရှင်သော်ဒါပဲလို့ ချက်ချင်းသိနိုင်တယ်။ သော်ဒါ ပထမတစ်ကြောင်း

| *"""Karel code goes here!"""*

ဟာ docstring လို့ ခေါ်တဲ့ စာသား ဖြစ်တယ်။ ဖန်ရှင်နဲ့ ပါတ်သက်တဲ့ ရှင်းလင်းဖော်ပြချက်တွေ ရေးဖို့ အတွက် သုံးတာပါ။ ဒါကြောင့် docstring ကို ချမှတ် သုံးခုတဲ့ *"""* တစ်စုံကြား ညုပ်ရေးတဲ့ ကွန်းမန်း တစ်မျိုးလို့ ယူဆနိုင်တယ်။ လိုအပ်တဲ့အခါ အသေးစိတ် ထပ်ပြီး ဖော်ပြပေးမှာပါ။

Docstring အောက်မှာ တွေ့ရတာကတော့ ကားချုပ်ကွန်းမန်းတွေဆိုတာ သိပါလိမ့်မယ်။ ကားချုပ်ကွန်းမန်းတွေဟာ stanfordkarel လိုက်ဘရဲ့ ဖန်ရှင်တွေပါ။ တနည်းအားဖြင့် stanfordkarel လိုက်ဘရဲ့မှာ ကားချုပ်ကွန်းမန်းတွေအတွက် ဖန်ရှင်သတ်မှတ်ချက်တွေ ပါဝင်တယ်။ ဖန်ရှင်တစ်ခုကို အသုံးပြုဖို့ အတွက် အဲဒီဖန်ရှင်ကို ခေါ်ပါတယ်။ ဒါကို function call ‘ဖန်ရှင်ကောလ’ လုပ်တယ်လို့ ပြောတယ်။ ကားချုပ်ကို ဘယ်ဘက်လုပ်စေချင်ရင် turn_left ဖန်ရှင်ကောလ လုပ်ရပါမယ်။ ဘိပါကောက်ခိုင်းချင်ရင် pick_beeper ဖန်ရှင်ကောလ လုပ်ရပါမယ်။ ဖန်ရှင်ကောလ လုပ်တဲ့ ပုံစံက

turn_left()
pick_beeper()

စသည်ဖြင့် ဖြစ်တယ်။

Python မှာ အင်ဒန်ထုတ်ကို ဖြစ်ကတတ်ဆန်း လုပ်လို့မရဘူး။ ဘေးမျဉ်းကနေ ဆွာတဲ့ အက္ဂာအဝေး မည်းတာနဲ့ ဆင်းတက်စေအမှား ဖြစ်တယ်။ မလိုတဲ့နေရာမှာလည်း ဆွဲရေးလို့ မရဘူး။ ခေါင်းစီးကို ဘေးမျဉ်းနဲ့ ဆွာထားကြည့်ပါ။ အယ်ရာဖြစ်တာကို တွေ့ရမယ်။ အင်ပိုဒ်တိတ်မန်းလည်း ဘေးမျဉ်းနဲ့ ကွာေနလို့မရဘူး။ အခြား language တွေမှားလည်း အင်ဒန်ထုတ်လုပ် ရေးကြပေမဲ့ Python မှာလောက် မတင်းကျပ်ဘူး။ အင်ဒန်ထုတ်မလုပ်လည်း ဆင်းတက်စေမှားတာ မဖြစ်ဘူး။

ကားချုပ်ပရှိရမ်တစ်ခုမှာ main ဟာ အထူးတာဝန်တစ်ခု လုပ်ဆောင်ပေးရတယ်။ အဲဒါကတော့ ပရှိရမ်ဝင်းဒီးမှာ Run Program ခလုတ် (ပဲ ၁.၅ မှာကြည့်ပါ) နှိပ်လိုက်ရင် တုံးပြန် လုပ်ဆောင်ပေးရတာပါ။ ဒါကြောင့် ကွန်းမန်းတွေဟာ အဲဒီခလုတ် နိုပ်တော့မှုပဲ စအလုပ်လုပ်တာ ဖြစ်တယ်။

Entry Point (ပရိုဂရမ်ဓမ္မတ်)

‘Meet Karel’ ပရိုဂရမ်မှာ main ဖန်ရှင်နောက် အောက်ခုံးအပိုင်းဟာ ပရိုဂရမ် run တဲ့အခါ ပထမဆုံး စတင်လုပ်ဆောင်ပေးရမဲ့ ဖန်ရှင်ကို ဖော်ပြပေးတာပါ။ ‘အန်ထရီပိုင့်’ လို့ခေါ်တယ်။

```
if __name__ == "__main__":
    run_karel_program("meet_karel")
```

run_karel_program ဖုန်ရှင်ဟာ ကားရဲပရိုကရမ် တစ်ခုအတွက် အန်ထရီပိုင့် ဖြစ်တယ်။ ပရိုကရမ် တက်လာတာနဲ့ တစ်ပါတော်း ခေါ်တင်ချင်တဲ့ ကဗ္ဗာကို ဒီဖန်ရှင်မှာ ထည့်ပေးတယ်။ `meet_karel.w` ကဗ္ဗာကို စေချင်းခေါ်တင်ထားချင်ရင် "meet_karel" ထည့်ပေးရမယ်။ ဖိုင်မရှိတဲ့ကဗ္ဗာကို ထည့်ထားရင် အယ်ရာတက်ပြီး ပရိုကရမ်ပွင့်လာမှာ မဟုတ်ဘူး။ ကဗ္ဗာမထည့်ပေးထားဘဲ ဒီလို

```
run_karel_program()
```

ဆိုရင် 8×8 အချယ် default ကဗ္ဗာကို တင်ပေးပါတယ်။

ကားရဲလ်ကဗ္ဗာတစ်ခုကို လိုချင်တဲ့ပုံစံ ဒီဇိုင်းဆဲပြီး ဖိုင်နဲ့ သိမ်းထားရတာပါ။ ကဗ္ဗာ ပုံစံချုတဲ့ ပရိုကရမ်လည်း ရှိတယ်။ ကဗ္ဗာဖို့တွေက .w ဖိုင် အိပ်စ်တန်းရှင်းနဲ့ ဖြစ်တယ်။ ဒီစာအုပ်မှာပါတဲ့ ဥပမာတွေလေ့ကျင့်ခန်းတွေ အားလုံးအတွက် လိုအပ်တဲ့ ကဗ္ဗာတွေကို အဆင်သင့်ပေးထားမှာပါ။ ကိုယ့်ဟာကို လုပ်ဖို့မလိုဘူး။ စိတ်ဝင်စားရင် စမ်းကြည့်လိုရအောင် စာမျက်နှာ (၃၇) နောက်ဆက်တဲ့ (ခ) မှာ အကျဉ်းဖော်ပြပေးထားပါတယ်။

၁.၃ ကားရဲလ် ပရိုကရမ် run ခြင်း

လိုအပ်တဲ့ဆောဖို့တွေ ထည့်သွင်းနည်းကို စာမျက်နှာ (၁၃) နောက်ဆက်တဲ့ (က) မှာ တစ်ဆင့်ချင်းဖော်ပြပေးထားပါတယ်။ အခုက Python ပရိုကရမ်တစ်ခုကို အရှုံးရှင်းဆုံး (လွယ်တယ်လို့ မဆိုလို) run လိုရတဲ့ နည်းကိုဖော်ပြပေးမှာပါ။ သဘောတရားပိုင်း နားလည်ဖို့ အထောက်အပံ့ဖြစ်မယ်။ အခုနည်းလမ်းကို အကြမ်းဖျော်း နားလည်အောင် ဖတ်ပြီးမှ နောက်ဆက်တဲ့ (က) ကို ဖတ်စေချင်ပါတယ်။

မိုက်ခရီးဆောဖို့ ဝင်းဒီးမှာ Notepad ၊ အက်ပလ် မက်ခံအက်စ်မှာTextEdit စတဲ့ တက်စ်အယ်ဒီတာတစ်ခုခဲ့နဲ့ ပရိုကရမ်ကုပ်ရေးလိုရတယ်။ ကုဒ်ဖိုင်ကို .py အိပ်စ်တန်းရှင်းနဲ့ သိမ်းရပါမယ်။ ပလိန်းတက်စ် (plain text) ဖိုင် ပါပဲ။ Python ကုဒ်ဖိုင်မှုလို .txt အတေား .py နဲ့ သိမ်းတာပါ။ Python ဖိုင် နံမည်ကို စာလုံးအသေးနဲ့ပဲ ပေးတဲ့ ထုတံ့ရှိတယ်။ စပော်နေရာမှာ _ (underscore) သုံးတဲ့ ထုံးစံရှိတယ်။ ဒါကြောင့် 'Meet Karel' ပရိုကရမ်ကုပ်ကို `meet_karel.py` ဖိုင်မှာ သိမ်းသင့်တယ်။

Python နဲ့ ရေးထားတဲ့ ပရိုကရမ်ကို run မယဆိုရင် Python ဆောဖို့ရှိရမှာပါ။ ဒီဆောဖို့တဲ့ အင်စတောလ လုပ်နည်းကို နောက်ဆက်တဲ့ (က) စာမျက်နှာ (၂၃) မှာ ဖော်ပြပေးထားပါတယ်။ Python ကုပ်တွေကို ကွန်ပျူးတာက တိုက်ရိုက် နားလည်ပါဘူး။ Python ဆောဖို့တဲ့ Python ကုပ်တွေကို တိုက်ရိုက် နားလည်ပြီး ကွန်ပျူးတာပေါ်မှာ run လိုရအောင် ကြားခံဆောင်ရွက်ပေးတဲ့ ဆောဖို့တဲ့ ယေဘုယျအားဖြင့် ယူဆနိုင်တယ်။

Python ထည့်ပြီးရင် stanfordkarel လိုက်ဘရိုက် အောက်ပါကွန်မန်းနဲ့ အင်စတောလ လုပ်ရပါမယ်။ အင်တာနက်ပေါ်ကနေ ဒေါင်းလှု လုပ်ရတာမျိုးလို့ ကွန်နက်ရှင်ရှိရမယ်။

```
pip install stanfordkarel
```

ကားရဲလ်ပရိုကရမ်မှာ ခေါ်တင်ချင်တဲ့ ကဗ္ဗာဖိုင်တွေလည်း ရှိရပါမယ်။ `meet_karel.w` ကဗ္ဗာဖိုင်က `meet_karel.zip` ဖိုင် worlds ဖို့ဒါထဲမှာ ရှိပါတယ်။ <http://tinyurl.com/3mmmm9c7j> လုပ်ကနေ `meet_karel.zip` ဖိုင်ကို ဒေါင်းလှုပ်လှုပ်ပါ။ ဒါ zip ဖိုင်ထဲက worlds ဖို့ဒါကို `meet_karel.py` ဖိုင်ရှိတဲ့ နေရာမှာ ကော်ပိုက်းထည့်ပါ။ ဝင်းဒီးမှာ Command Prompt ၊ မက်ခံအက်စ်မှာ Terminal ဖွင့်ပြီး `cd` ကွန်မန်းနဲ့ ကုဒ်ဖိုင်ထားတဲ့ ဖို့ဒါထဲကို သွားပြီး အောက်ပါအတိုင်း `python` ကွန်မန်းနဲ့ ကုဒ်ဖိုင်ကို run ပေးရပါမယ် (လာမဲ့စာမျက်နှာမှာ နမူနာပြထားတာ ကြည့်ပါ)။

```
python meet_karel.py
```

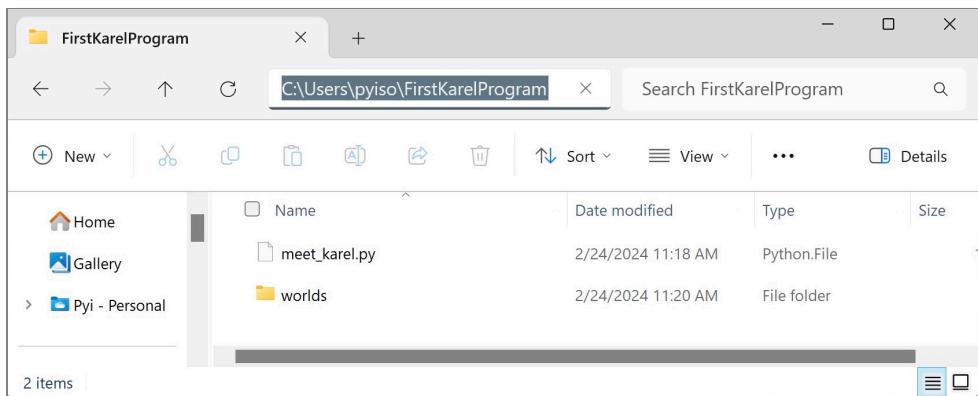
ကုဒ်ရေးထားတာ ဆင်းတက်စာများ မရှိဘူးဆိုရင် ကားရဲပရိုကရမ် ပွင့်လာမှာပါ။

အခြောက်ပြုခဲ့တာက ကားရဲလ်ပရိုကရမ်တစ်ခု run ဖို့ မဖြစ်မနေလုပ်ရမှာ အနည်းဆုံးလိုအပ်ချက်ပါ။ Python ဆော်ဖို့ ရှိရမယ်။ stanfordkarel လိုက်ဘရီ အင်စတောလ် လုပ်ရမယ်။ ကဗျာဖိုင်ပါတဲ့ worlds ဖို့ရှိရမယ်။ .py ဖိုင် တစ်ခုနဲ့ ပရိုကရမ်ကုဒ်ကို သံမ်းရမယ်။ worlds ဖို့ဒါနဲ့ ကုဒ်ဖိုင်ကို တစ်နေရာတည်းမှာ ထားရမယ်။ ပြီးရင် ကွန်မန်းလိုင်းမှာ

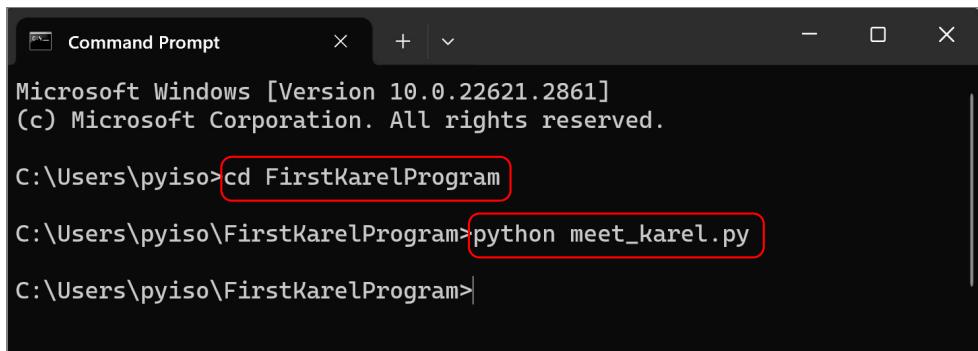
```
python your_karel_program.py
```

run ရဲပါပဲ။

C:\Users\pyiso\FirstKarelProgram ဖို့ထဲမှာ ကုဒ်ဖိုင်နဲ့ worlds ဖို့ကို ထားပြီး ဘယ်လို run ရလဲ နမူနာပြထားတာကို ကြည့်ပါ။



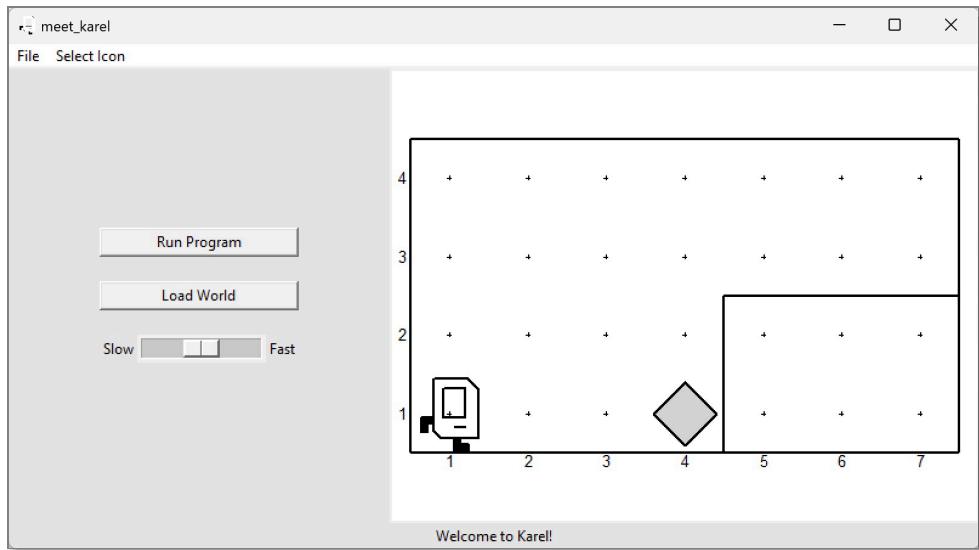
ပုံ ၁.၃



ပုံ ၁.၄

၁.၄ Move Beeper to Other Side

ပရိုကရမ်းမင်း လေ့လာတဲ့အခါ စာချည်းပဲ ဖတ်နေပြီး အမှန်တကယ် နားလည်သွားဖို့ဆိုတာ မဖြစ်နိုင်ပါဘူး။ လက်တွေ့ စမ်းသပ်ကြည့်၊ ရေးကြည့်မှာ တကယ် နားလည်လာမယ်။ တကယ်လည်း ကျမ်းကျမ်းကျင်ကျင်

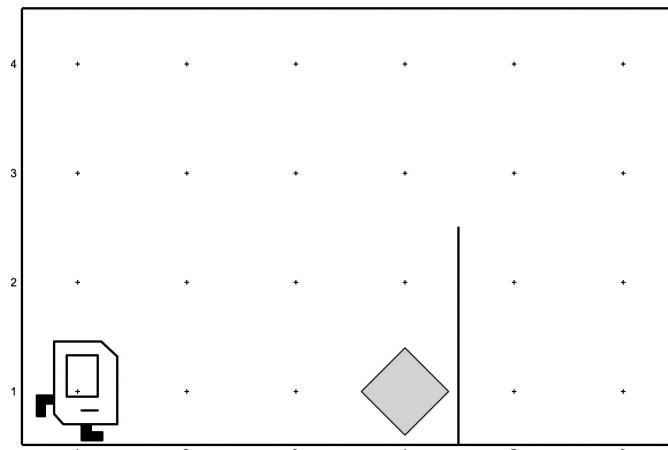


ပုံ ၁.၅

ရေးတတ်လာမှုပါ။ ဒါကြောင့် လက်တွေ့ရေးကြည့်ပါ။ များများ လေ့ကျင့်ပါ။ ဥပမာတွေကိုလည်း နားလည်အောင် ဖတ်ပြီးရင် မိမိဘာသာ အလွတ် ပြန်ရေးကြည့်ပါ။

ပုံ (၁.၆) မှ ဘိပါကို နံရအဥားတစ်ဘက် အောက်ခြေကို ရွှေ့ပေးတဲ့ ပရိုဂရမ် ရေးကြည့်ပါ။ Python ထုံးစာရွက် `move_beeper_to_other_side.py` ဖိုင်နဲ့ သိမ်းသင့်ပါတယ်။ `meet_karel.zip` ဖိုင်ထဲမှုပါတဲ့ worlds ဖို့မှာပဲ အခု ကမ္ဘာဖိုင် ထည့်ပေးထားပါတယ်။ `move_beeper_to_other_side.w` နံမည့်နဲ့ပါ။ အန်ထရိုပိုင့်အတွက် အခုလိုရေးရပါမယ်။

```
if __name__ == "__main__":
    run_karel_program("move_beeper_to_other_side")
```



ပုံ ၁.၆

ଓପଣ୍ଡିତ
ଜ

ଗୁରୁତ୍ବାଳେ ମହାନ୍ତିମ୍ଭାଗ

•J

နောက်ဆက်တဲ့ က

လိုအပ်သည့် ဆောဖံ့ဌများ ထည့်သွင်းခြင်း

အခြား အင်ဂျင်နီယာ/သိပ္ပံဌ ပညာရပ်တွေလိုပဲ ပရိုကရမ်မင်းလေ့လာတဲ့အခါ လက်တွေ့လုပ်ကြည့်ဖို့ အင်မတန်အရေးကြိုးပါတယ်။ လက်တွေ့ရေးမကြည့်ဘဲ စမ်းသပ်မကြည့်ဘဲ သဘောတရားပိုင်းဆိုင်ရာတွေကို အမှန်တကယ်နားလည်ဗူး မဟုတ်ပါဘူး။ အခြားပညာရပ်တွေထက် ကွန်ပျိုးတာ ပရိုကရမ်မင်းရဲ့ အားသာချက်တစ်ခုကတော့ လက်တွေ့စမ်းသပ်ခန်းကြိုးတွေ ရှိစရာမလိုတာပါပဲ။ စမ်းသပ်ပစ္စည်းတွေ လည်း များများစားစား မလိုအပ်ဘူး။ ကွန်ပျိုးတာ တစ်လုံးနဲ့ လိုအပ်တဲ့ ဆောဖံ့ဌပဲတဲ့ ရှိရင်ရပြီ။ ဆောဖံ့ဌတွေကလည်း ပိုက်ဆံကုန်စရာမလိုဘူး။ မပေးဘဲ သုံးလို့ရတာ။

ပရိုကရမ်လက်တွေ့ရေး လေ့ကျင့်ဖို့အတွက် လိုအပ်တဲ့ ဆောဖံ့ဌတွေ ထည့်ထားရပါမယ်။ Python programming language အတွက် Python ဆောဖံ့ဌ ထည့်ရပါမယ်။ Python ဆောဖံ့ဌမရှိဘဲ Python ကုဒ်တွေ၊ Python ပရိုကရမ်တွေ run လို မရပါဘူး။ Python အပြင် ပရိုကရမ်ကုဒ် ရေးဖို့ အတွက် အထောက်အကူဗြာ ဆောဖံ့ဌပဲတစ်ခုလည်း လိုအပ်တယ်။

ပရိုကရမ် ကုဒ်ရေးဖို့အတွက် PyCharm သိမဟုတ် Visual Studio Code (VS Code) ကို အသုံးပြုနိုင်ပါတယ်။ အက်ဆေးတစ်ပုဒ်ရေးတဲ့အခါ မိုက်ခရိုဆောဖံ့ဌ Word နဲ့ရေးလိုရသလို ရိုးရိုးရှင်းရှင်း Notepad လောက်နဲ့ ရေးလိုလည်း ရတာပါပဲ။ အက်ဆေးရဲ့ အဓိပ္ပာယ်ကသာ အစိုကပါ။ ပရိုကရမ်ကုဒ် ရေးတဲ့အခါမှာလည်း ခီးသောပါပဲ။ ဆောဖံ့ဌပဲရေးကြိုးတွေ တည်ဆောက်တဲ့အခါ လိုအပ်မဲ့ ဖီချာတွေ အားလုံး စုံစုံလင်လင်ပါပြီးသား PyCharm လို Integrated Development Environment(IDE) ဆောဖံ့ဌပဲများနဲ့ ကုဒ်တွေရေးလိုရသလို ပါ့ပေါ့ပါ့ပါ့နဲ့ လိုအပ်မှုပဲ လိုတဲ့ဖီချာအတွက် extensions (plugin လိုလည်းခေါ်ယ်) ထည့်သွင်းရတဲ့ Visual Studio Code (VS Code) လို ကုဒ်အသိဒ်ဘာ (Code Editor) ဆောဖံ့ဌပဲ့ သုံးပြီး ရေးရင်လည်း ရတာပါပဲ။ နောက်ဆုံး ကုဒ်နည်နည်း၊ ဖိုင်နည်းနည်း ရိုးရှင်းတဲ့ ပရိုကရမ်လေးတွေဆုံးရင် Notepad နဲ့ ရေးလိုတောင် ရပါတယ်။ ကုဒ်တွေများမယ် ဖိုင်တွေ များမယ်၊ အသင်းအဖွဲ့လိုက် ပူးပေါင်းရေးရတဲ့ ပရိုကရမ်မျိုးတွေ ဆိုရင်တော့လည်း Notepad လောက်နဲ့ အဆင်မပြနိုင်တော့ဘူးပေါ့။

PyCharm ရော VS Code ထည့်သွင်းနည်းပါ ဖော်ပြပေးပါမယ်။ မိမိ နှစ်သက်ရာ အဆင်ပြောရာ သို့မဟုတ် နီးစပ်ရာ ပရိုကရမ်ဘာ အသိမိတ်ဆွေ အကြုပြုတဲ့ တစ်ခုကို ရွေးချယ်သုံးပါ။ စလေ့လာသူအနေနဲ့ PyCharm ကို အသုံးပြုတာ ပိုအဆင်ပြောမယ်လို ထင်တယ်။ PyCharm သုံးကြည့်လို မိမိကွန်ပျိုးတာ မှာ နေးလွန်းတယ်ဆိုရင် VS Code ကိုစမ်းကြည့်ပါ။ နှစ်ခုလုံး စက်အရမ်းကောင်း/မြင့် ဖို့ မလိုပါဘူး။ တော်ရုံး အတန်အသင့်ကောင်းတဲ့ စက်လောက်နဲ့ အဆင်ပြောပါတယ်။

မိမိကိုယ်တိုင်က ကွန်ပျိုးတာ အသုံးပြုပဲ အခြေခံ အားနည်းပြီး ဖော်ပြပေးထားတဲ့ အတိုင်း တစ်ဆင့် ချင်း အင်စတောလ် လုပ်ဘာလည်း အဆင်မပြောဖြစ်နေရင် ဒီဘာအုပ်ရဲ့ အောက်ပါ ဖွေ့စွဲဘွဲ့တ်ခဲ့နဲ့ ယူကျ။

ချုပ်နယ်တွေမှာ ကြည့်ရှုမေးမြန်း အကူအညီ တောင်းဆိုင်ပါတယ်။ ဒါမှမဟုတ် အတွေးအကြိုးစိတ် နဲ့ စီးပွားရေးမှုပြီး အင်စတောလ်လုပ်ပါ။

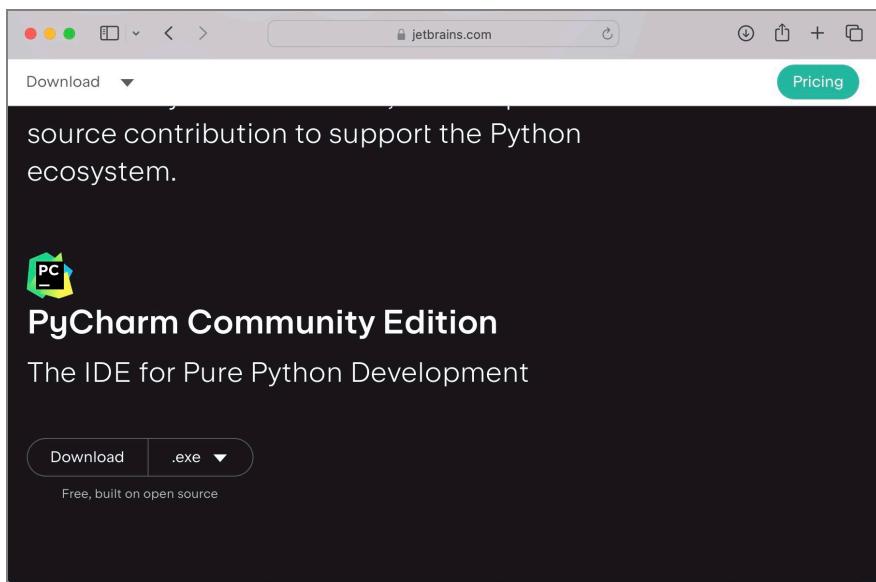
<https://www.facebook.com/bpwp>

<https://www.youtube.com/bpwp>

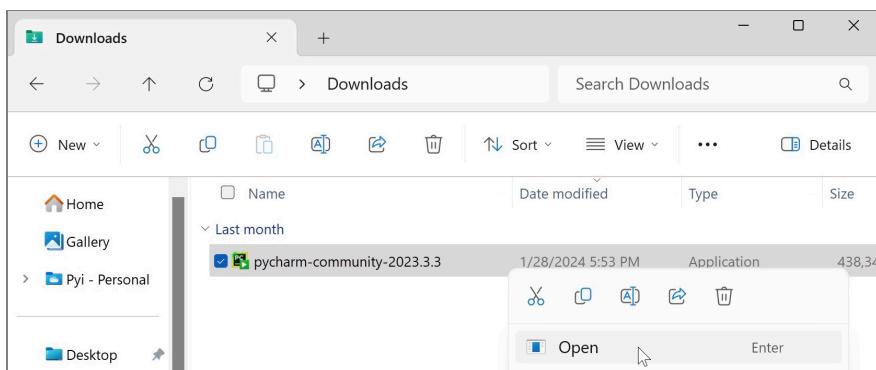
PyCharm အတွက် အရင်ဖော်ပြပေးပါမယ်။ VS Code အတွက် စာမျက်နှာ ၂၆ မှာ ကြည့်ပါ။

Python နှင့် PyCharm IDE ထည့်သွင်းခြင်း

<https://www.jetbrains.com/pycharm/download/> လင့်ကိုဖွင့်ပါ။ ဝဘ်စာမျက်နှာ အောက်ဘက် နည်းနည်း ဆွဲချက်ရင် PyCharm Community Edition ခေါင်းလုပ်ခလုပ်ကို တွေ့ရပါမယ်။ ပုံ (၁/၁) ကိုကြည့်ပါ။



ပုံ ၁/၁



ပုံ ၃/၂

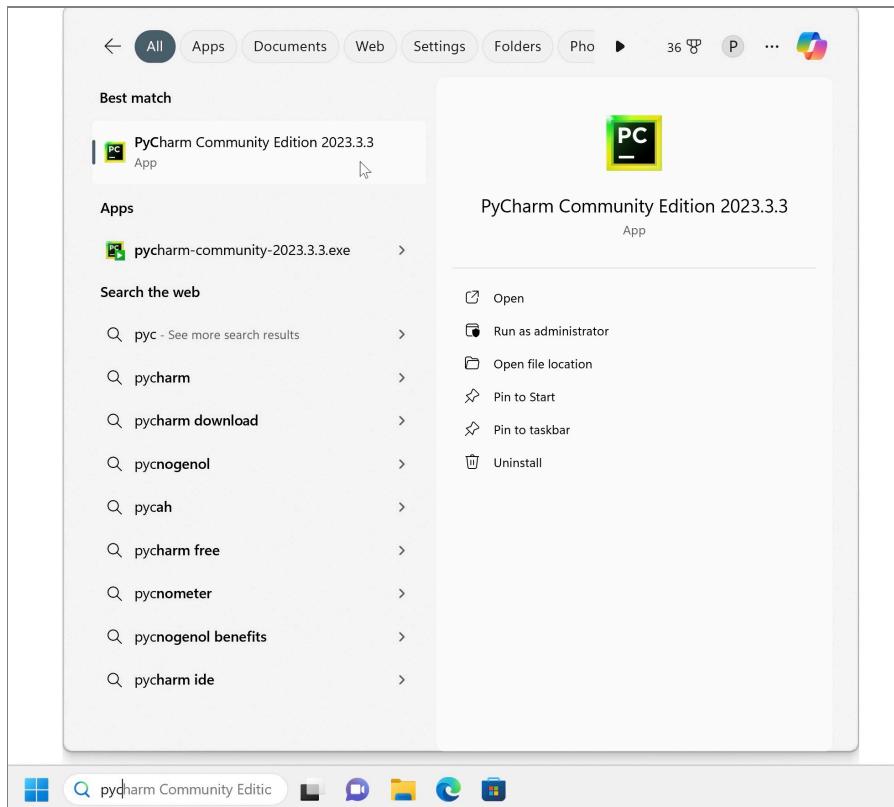
PyCharm Community Edition ကို အောင်လုပ်လုပ်ပါ။ (ဝဘ်စာမျက်နှာ အပေါ်ပိုင်းက ၀၂၍ သုံးရတဲ့ PyCharm Professional ကို အောင်လုပ် မှားမလုပ်မိဖို့ သတိပြုပါ)။ အင်စတော်လာဖိုင်ကို ညာကလစ်နှုပ်ပြီး Open လုပ်ပါ (ပုံ ၃/၂ ကိုကြည့်ပါ)။ Yes/No မေးတဲ့အခါ Yes နှုပ်ပါ။ Next > ကို နှုပ်၍ ရွှေ့ကိုဆက်သွားပြီး နောက်ဆုံးမှာ Install နှုပ်ပြီး ကွန်ဖန်လုပ်ပါ။ အင်စတော်ပြီးသွားရင် Finish နှုပ်ပါ။

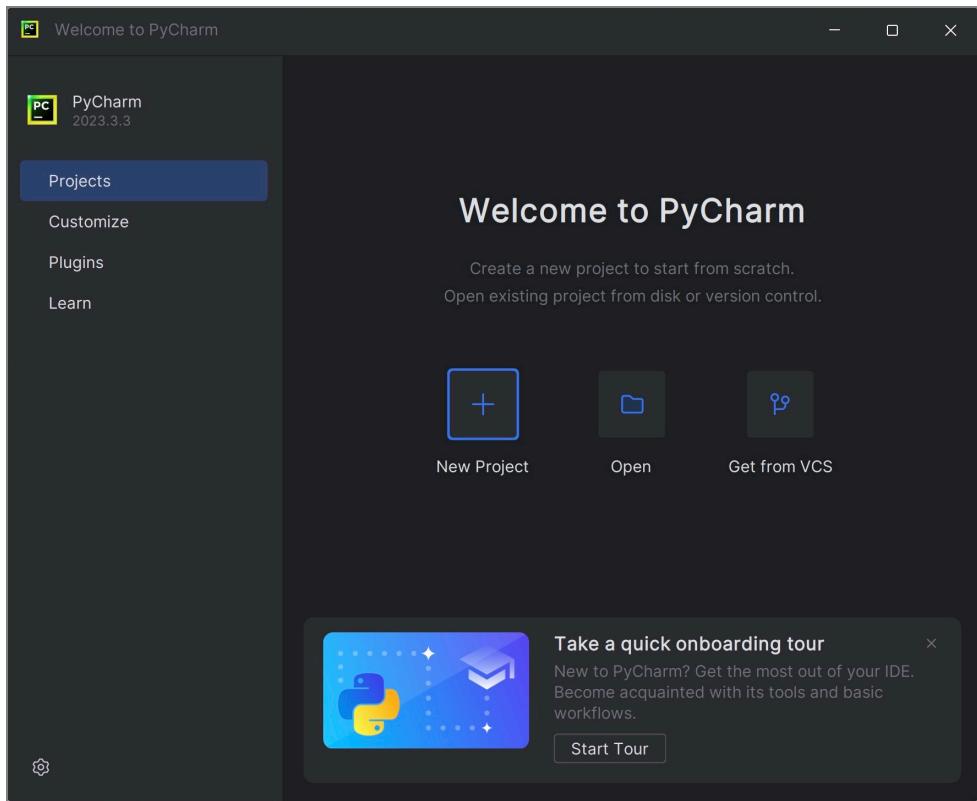
၀၂၃: Taskbar Search ကနေ PyCharm ကိုရှုပြီးဖွံ့ဖြိုးပါ (ပုံ ၃/၃ ကို ကြည့်ပါ)။ သဘောတူ ကြောင်း ကွန်ဖန်လုပ်ခိုင်းရင် ချက်ချက်ဘောက်စ် ချက်ချက်လုပ်ပြီး Continue နှုပ်ပါ။ ဒေတာပို့ချင်လား ထပ် မေးပါလိမ့်မယ်။ Don't Send နှုပ်ပါ။ Welcome စာရင်ကိုပေါ်လာမယ်။ ပုံ (၃/၄) မှာ ကြည့်ပါ။

ဒီစာရေးနေချိန် လက်ရှိ PyCharm ဗားရှင်းက ၂၀၂၃ ပါ။ သိပ်မကြာခင် ၂၀၂၄ ထွက်ပါတော့ မယ်။ အကယ်၍ လက်ရှိဗားရှင်းထက် နိမ့်တဲ့ဟားရှင်းတွေကို အောင်လုပ် လုပ်ချင်ရင် အောက်ပါ လင့်ကို သွားပါ။

<https://www.jetbrains.com/pycharm/download/other.html>

ဗားရှင်း ၂၀၂၄/၂၅ ထွက်ပြီးတဲ့ အချိန်မှာ ၂၀၂၃ ဗားရှင်းကို လိုချင်ရင် ဝဘ်စာမျက်နှာမှ ရှာပြီး အောင်လုပ် လုပ်ပါ။ ၂၀၂၃ မှာလည်း ဗားရှင်းအခွဲတွေ ရှိပါသေးတယ်။ လက်ရှိအမြင့်ဆုံး ဗားရှင်းအခွဲ (ဥပမာ ၂၀၂၃.၃.၃) ကို သုံးလိုပါတယ်။ ၂၀၂၄/၂၅ သုံးမယ်ဆုံးရင်လည်း ပြဿနာတော့ မရှိပါဘူး။ Update ဗားရှင်းဖြစ်တဲ့အတွက် ပုံတွေမှာ ပြထားတာနဲ့တော့ ကွာ့ခြားချက်တရာ့၊ ရှိကောင်းရှိနိုင်ပါတယ်။

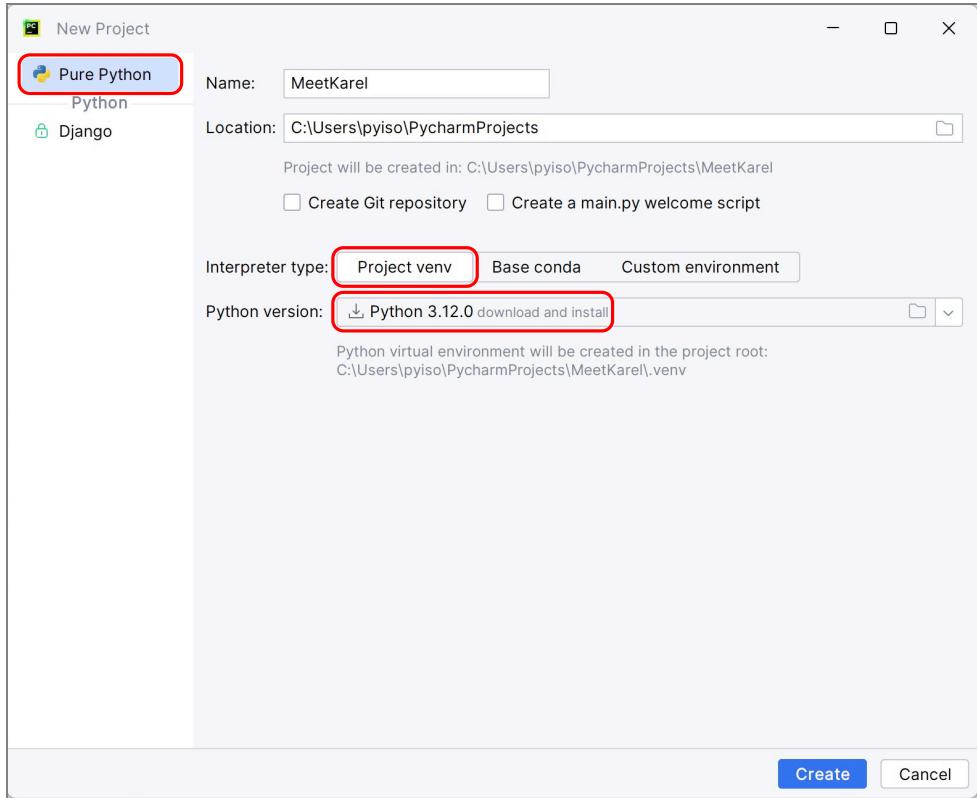




ပုံ ၂/၄

PyCharm IDE ဆိတာဘာလ

PyCharm ဟဲ Python နဲ့ ဆောင်ဝဲ ရေးဖိုအတွက် အထောက်အကူဗြှု Integrated Development Environment(IDE) ဆောင်ဝဲဖြစ်ပါတယ်။ စာစီတရိက်လုပ်တဲ့အခါ Microsoft Word ကို အသုံးပြု ကြသလိုပဲ Python ကုဒ်ရေးဖို PyCharm ကိုထံးတဲ့ သဘောပေါ့။ PyCharm IDE ၏ Python ပရောဂျက်တွဲ အတွက် အခံကရည်ရွယ်တယ်။ အထောက်အဦးတစ်ခု၊ တံတားတစ်ခု ဆောက်လုပ်တာ ကို ပရောဂျက်လို့ ပြောလေ့ရှိသလို ပရိုကရမ်/ဆောင်ဝဲတစ်ခု တည်ဆောက်တာကိုလည်း ပရောဂျက်လိုပဲ သုံးနှုန်းပါတယ်။ တစ်ဦးတစ်ယောက်တည်း ရေးတဲ့ ပရိုကရမ်အသေးလေးတွဲ အတွက် PyCharm ကို အသုံးပြုနိုင်သလို ပရိုကရမ်မာတွေ အဖွဲ့လိုက်နဲ့ တည်ဆောက်ရတဲ့ ပရောဂျက်ကြီးတွဲ အတွက်လည်း သုံးပါတယ်။ ဒီစာအုပ်မှာတော့ PyCharm ရဲ့ အဆင့်မြင့်ဖိုချာတွေကို အသုံးပြုမှာ မဟုတ်ပါဘူး။ ပရိုကရမ်း မင်း စလေ့လာသူတွေကို လွှာယ်ကူးအဆင်ပြေစေတဲ့ အခြေခံ ဖိုချာတွေလောက်ပဲ အသုံးပြုမှုပါ။



ပုံ ၂/၅ ပရောဂျက် အသစ်ယူခြင်း

PyCharm ပရောဂျက်ဆောက်ခြင်း

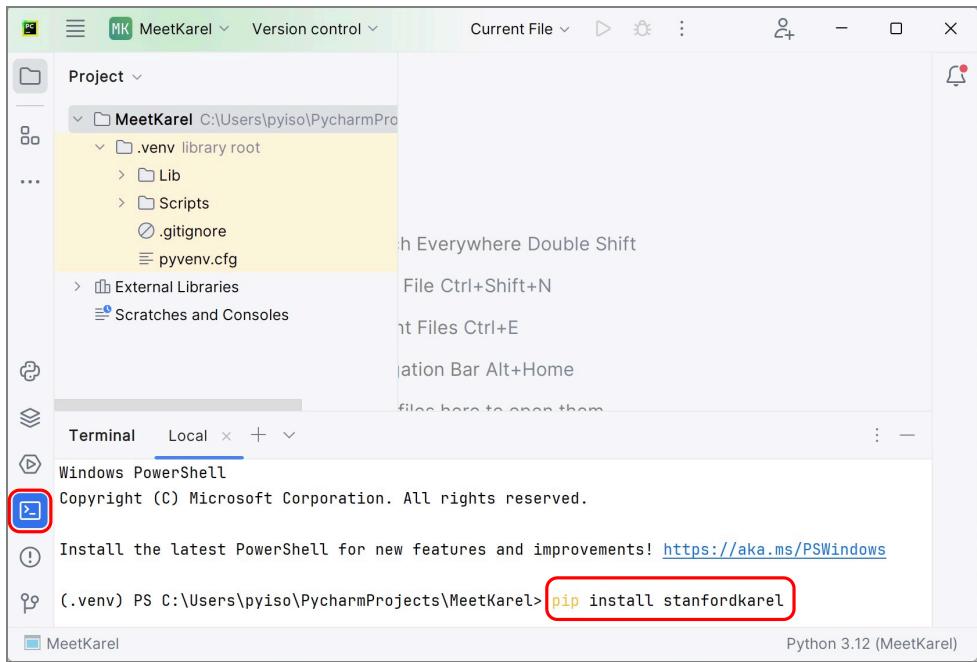
အင်စတောလုပ်ပြီးရင် PyCharm IDE ကိဖွဲ့ပြီး Welcome စခရင်မှ သိမဟုတ် File မိန္ဒီမှ New Project နှိပ်ပြီး ပရောဂျက် အသစ်ယူပါ (ပြီးခဲတဲ့ ပုံ ၂/၅ ကို ပြန်ကြည့်ပါ)။ နံမည်ကို MeetKarel သေးပါ။ ပုံ (၂/၅) မှာ တွေ့ရတဲ့အတိုင်း Pure Python, Proj venv နဲ့ Python ဘားရှင်း 3.12.xx ကို ရွေးပါ။ xx က အခြားကဏ္ဍား ဖြစ်နေနိုင်တယ်။ အခိုက် ဘားရှင်း 3.12 သာ ဖြစ်ပါပေါ်။ Create ခလုတ်နှုပ်ပါ။ ပရောဂျက်အတွက် Python ကို အင်စတောလုပ်ပါလိမ့်မယ်။

မိမိလေ့လေနေတဲ့ အခန်းတစ်ခန်းချင်းစီအတွက် ပရောဂျက်တစ်ခု ဆောက်နိုင်ပါတယ်။ အခန်း (၂) အတွက် Chapter02၊ အခန်း (၃) အတွက် Chapter03 စသည်ဖြင့်။ သက်ဆိုင်ရာအခန်းအလိုက် ကုဒ်ဖိုင်တွေကို ပရောဂျက်တစ်ခုစီမှာ ထားတဲ့အတွက် ဖိုင်တွေများပြီး ရှုပ်ထောင်းပွဲနေတဲ့ ပြဿနာ မရှိတော့ဘူး။ ကုဒ်ဖိုင်တွေကို ပရောဂျက်တစ်ခုလဲမှာပဲ ဖိုဒ်အလိုက်ခဲ့ထားလို့ရပေမဲ့ စလေ့လာသူအနေနဲ့ ပရောဂျက်တစ်ခုချင်း ခဲ့ထားတာလောက် မလွယ်ကူဘူး။

ပရောဂျက် အသစ်ယူတဲ့အခါ တည်နေရာ Location: ကို သူ့ရှိအတိုင်း ထားနိုင်သလို မိမိထားချင်တဲ့နေရာကို ညာဘာက်စွန်း ဖိုဒ်အိုင်ကွန်လေးနှိပ်ပြီး ပြောင်းလို့ရတယ်။

stanfordkarel လိုက်ဘရီ အင်စတောလုပ်ခြင်း

ပုံ (၂/၆) မှာ အနီရောင် ဝိုင်းပြထားတဲ့ အိုင်ကွန်ကို နှိပ်ပြီး Terminal ကိဖွဲ့ပါ။ Terminal မှာ အောက်ပါ ကွန်မန်းဖြင့်



ပုံ ၂/၆ Karel လိုက်ဘရီ အင်စတောလ်လုပ်ခြင်း

```
pip install stanfordkarel
```

ကားရဲ့လိုက်ဘရီကို အင်စတောလ်လုပ်ပါ။ ပုံ (၂/၆) မှာ အနီးဆုံး ပိုင်းပြထားပါတယ်။ ခက္ကာတဲ့ အခါ အခုလို မက်ဆွဲချုပ်တွေ ကျလာပါလိမ့်မယ်။

```
Collecting stanfordkarel
  Downloading stanfordkarel-0.2.7-py3-none-any.whl (51 kB)
  eta 0:00:00
    51.9/51.9 kB 443.1 kB/s
Installing collected packages: stanfordkarel
Successfully installed stanfordkarel-0.2.7
```

```
[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
(.venv) PS C:\Users\pyiso\PycharmProjects\MeetKarel>
```

ဟိုက်လိုက်ပြထားတဲ့ မက်ဆွဲချုပ်တွေရရင် အင်စတောလ်လုပ်တာ အောင်မြင်လိုပါ။

မှတ်ချက်။ ပရောဂျက် အသစ်ဆောက်တိုင်း လိုအပ်တဲ့ လိုက်ဘရီကို တစ်ခါတပ်ပြီး အင်စတောလ် လုပ်ရပါမယ်။ ကားရဲ့အခန်း ပရောဂျက်တစ်ခုစီအတွက် stanfordkarel လိုက်ဘရီကို အထက်ဖော်ပြပါအတိုင်း အင်စတောလ် လုပ်ဖို့လိုတယ်။

နမူနာ ကားရဲလ် ကဗ္ဗာနှင့် ပရိဂရမ်ကုဒ် ဖိုင်များထည့်ခြင်း

meet_karel.zip ဖိုင်ကို ဒီလင့် <http://tinyurl.com/3mmmm9c7j> ကနေ ဒေါင်းလုပ်လုပ်ပါ။ ငါး zip ဖိုင်ကို extract လုပ်ပါ။ meet_karel နံမည်နဲ့ ဖိုဒါတစ်ခု ရလာပါမယ်။ ငါးဖိုဒါထဲမှ အောက်ပါ worlds ဖိုဒါနှင့် .py ဖိုင်အားလုံးကို ကော်ပီလုပ်ပါ။

- worlds
 - ▶ meet_karel.w
 - ▶ move_beeper_to_other_side.w
- meet_karel.py
- move_beeper_to_other_side.py
- world_editor.py

MeetKarel ပရောဂျက်ထဲတွင် ကူးထည့်ပါ။ ပင်မ ပရောဂျက် MeetKarel (ပုံ က/၇ မှာ မြှားပြထား) ပေါ်မှာ ညာကလစ်နှုပ်ပြီး Paste လုပ်ရမှာပါ။ ကော်ပီကူးထည့်လိုက်တဲ့ ဖိုင်တွေက ပုံမှာ တွေ့ရတဲ့ အတိုင်း MeetKarel ဖိုဒါအောက်မှာ ရှိသင့်ပါတယ်။

အခန်းအလိုက် နမူနာ ကုဒ်ဖိုင်တွေ ထည့်ပေးထားတဲ့ .zip ဖိုင်တွေကိုလည်း အထက်ပါအတိုင်း အလားတူ လုပ်ရပါမယ်။ ပရောဂျက်အသစ်ဆောက်၊ .zip ဖိုင်ကို ဖြည့်၊ ရလာတဲ့ ဖိုဒါထဲက ဖိုင်တွေကို ပင်မ ပရောဂျက် ဖိုဒါထဲ ကော်ပီကူးထည့် ရုံပါပဲ။

meet_karel.py ဖိုင်ကို ကလစ်နှစ်ချက်နှုပ် ဖွင့်ပါ။ ပုံ (က/၇) မှာ အနီးပိုင်းထားတဲ့ ကုဒ်အယ်ဒီတာ (code editor) ပွင့်လာပါမယ်။ အဲဒီ မူရင်းဖိုင်မှာ အောက်ပါအတိုင်း ဆက်လက် ပြင်ဆင်ဖြည့်စွက်ပါ။

```
from stanfordkarel import *

def main():
    """Karel code goes here!"""
    move()
    move()
    move()
    pick_beeper()
    turn_left()
    move()
    move()

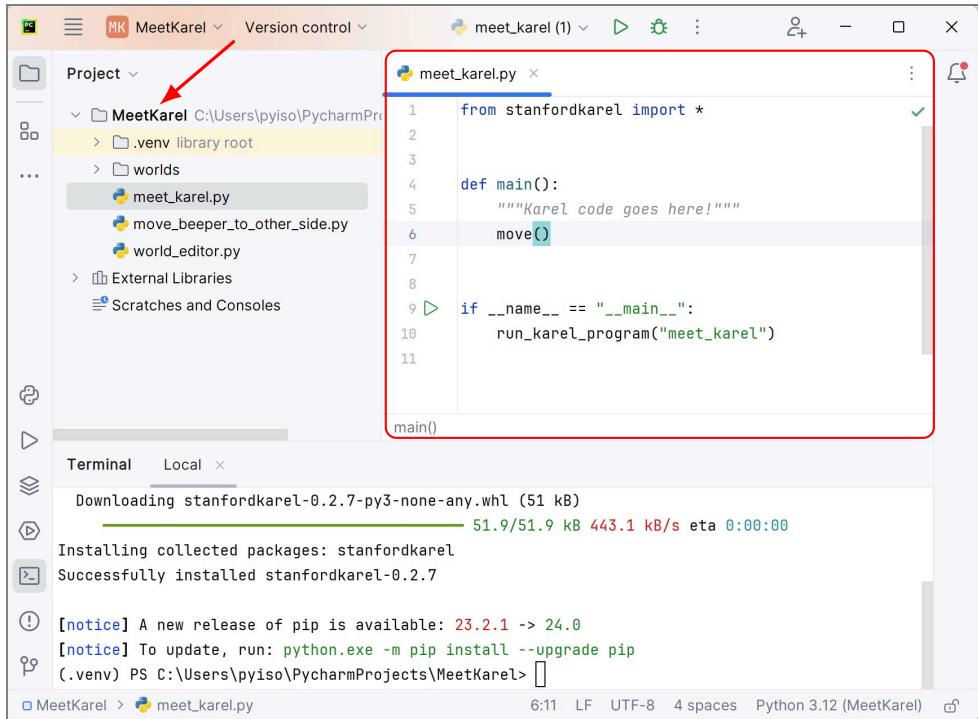
    turn_left()
    turn_left()
    turn_left()

    move()
    put_beeper()

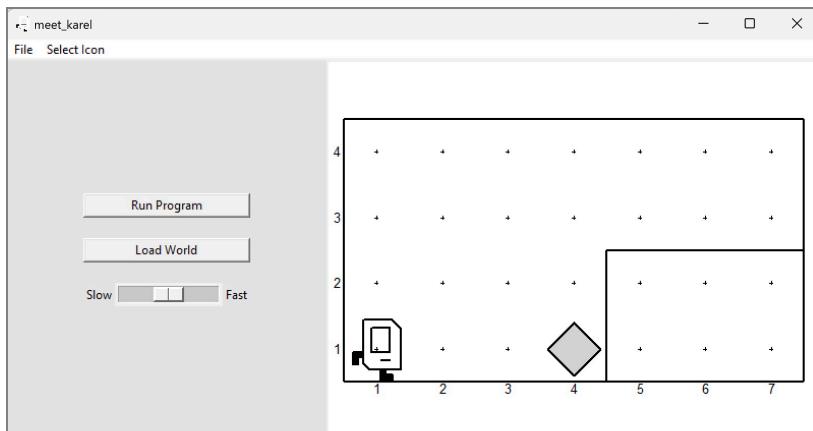
if __name__ == "__main__":
    run_karel_program("meet_karel")
```

run ထားတဲ့ ပရိုကရမ်တစ်ခုကို တစ်ခါထပ် run ရင် Cancel (သို့) Stop and Rerun လုပ်မှာ လူမေးတယ်။ Stop and Rerun လုပ်ရပါတယ်။

meet_karel.py ဖိုင်ကို ညာကလစ်နှင့်ပြီး Run 'meet_karel' လုပ်ပါ (အယ်ဒီတာ ဧရိယာမှာ ညာကလစ်နှင့်ပြီး Run 'meet_karel' လုပ်လိုလည်း ရပါတယ်) ။ ပုံ (က/၈) က ကားရဲ့ပုံ ပရိုကရမ် တက်လာရင် Run Program ခလုတ်ကိုနိုပ်ပါ။ ဘိပါကို ကားရဲ့လ်က ရွှေပေးပါလိမ့်မယ်။



ပုံ က/၇



ပုံ က/၈

ဆင်းတက်စံအမှားများ

အကယ်၍ ပရိုက်မဲ့ run ပေါ်လို့မရရင် ကုဒ်ရေးတာမှားနေလို့ ဖြစ်နိုင်တယ်။ မမိရေးထားတာကို စာမျက်နှာ (၁၉) က ပရိုက်မဲ့ကုဒ်နဲ့ နှင့်ယူဉ် စစ်ဆေးကြည့်ပါ။ PyCharm အယ်ဒီတာမှာ အနိုင်တွန်းလေးတွေ (ပုံ ၃/၉) ပြတဲ့နေရင် အဲဒီနေရာတွေမှာ ဆင်းတက်စံမှားနေလို့ (သို့) လိုက်ဘရှိမထည့်ရသေးလို့ပဲ။

လိုက်ကွင်းကျို့နေတာက အဖြစ်များတဲ့ အမှားပါ။ ကျို့ခဲ့လို့ မရပါဘူး။ အင်ဒန်တေးရှင်း (indentation) လုပ်ရမဲ့နေရာမှာ မလုပ်ထားရင်လည်း ပြဿနာဖြစ်တယ်။ move, turn_left တွေကို ဘေးမျဉ်းလာဘက်ဆွဲပြီး အင်ဒန်လုပ်ပေးရမယ်။ အဲဒါတွေ ကရာမစိုက်မိရင် ဆင်းတက်စံအမှားဖြစ်ပြီး ပရိုက်မဲ့ run လို့ မရနိုင်ဘူး။

Terminal မှာ ထုတ်ပေးတဲ့ မက်ဆွဲချုပေါ်တွေကို ကြည့်ပြီးတော့လည်း ဘာပြဿနာဖြစ်နေလဲ မှန်းဆ လို့ရနိုင်တယ်။ ဘာကြောင့်ဖြစ်နိုင်လဲ ဆက်စပ်စဉ်းစားလို့ ရတယ်။ ဥပမာ ဖြစ်တဲ့ပြဿနာအလိုက် အခုလို တွေ့ရပါမယ်။

```
File "c:\Users\pyiso\VS Code\meet_karel\meet_karel.py", line 6
    move(
        ^
SyntaxError: '(' was never closed

File "c:\Users\pyiso\VS Code\meet_karel\meet_karel.py", line 7
    move()
        ^
IndentationError: unexpected indent

Traceback (most recent call last):
  File "c:\Users\pyiso\VS Code\meet_karel\meet_karel.py", line 1,
    in <module>
      from stanfordkarel import *
ModuleNotFoundError: No module named 'stanfordkarel'
```

```
File "c:\Users\pyiso\PycharmProjects\MeetKarel\meet_karel.py", line 6
    move(
        ^
SyntaxError: '(' was never closed

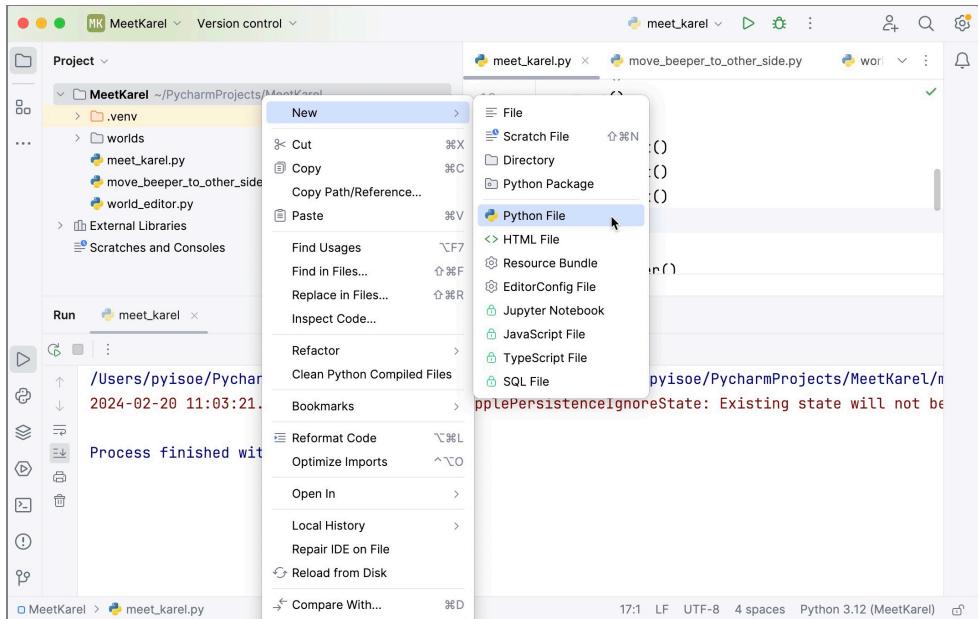
File "c:\Users\pyiso\PycharmProjects\MeetKarel\meet_karel.py", line 7
    move()
        ^
IndentationError: unexpected indent

Process finished with exit code 1
```

Python ဖိုင် အသစ်ယူခြင်း

MeetKarel ပင်မ ပရောဂျက်ဖိုဒီပေါ်မှာ ညာကလစ်နှုပ်ပြီး Python ဖိုင် အသစ်ယူနိုင်ပါတယ်။ Python ဖိုင်တွေက .py အိုပ်စံတန်းရှင်းနှုပါ။ ကားရဲလ်ပရိုဂရမ်တစ်ခုကို Python ဖိုင်တစ်ခု ထားပါမယ်။ ပင်မ ပရောဂျက်ဖိုဒီ အောက်မှာပဲ တိုက်ရိုက်ရှိပါမယ်။

နောက်ပိုင်း အဆင့်မြင့်လာရင် ပရိုဂရမ်တစ်ခုအတွက် ပရောဂျက်တစ်ခု ထားနိုင်တယ်။ ကုဒ်ဖိုင်တွေ အပြင် ပရိုဂရမ်အတွက် လိုအပ်တဲ့ ရုပ်ပုံတော့၊ အခြားဖိုင်တွေ (config ဖိုင်၊ setting ဖိုင် စသည်ဖြင့်) လည်း ပါနိုင်တယ်။ ပင်မပရောဂျက် အောက်မှာပဲ ဖိုင်တွေက တိုက်ရိုက်ရှိဖို့လည်း မလိုတော့ဘူး။ ဆက် စပ်ရာ ဖိုင်တွေကို အမျိုးအစားအလိုက် ဖန်ရှင်အလိုက် ဖို့ဒါတွေခဲ့ပြီး စနစ်ကျ စီစဉ်ဖွဲ့စည်း ထားရမှာပါ။ ပရောဂျက်တစ်ခုမှာ ဖိုင်တွေကို စနစ်တကျ စုဖွဲ့ထားဖို့ အရေးကြီးပါတယ်။



ပုံ က/၁၀

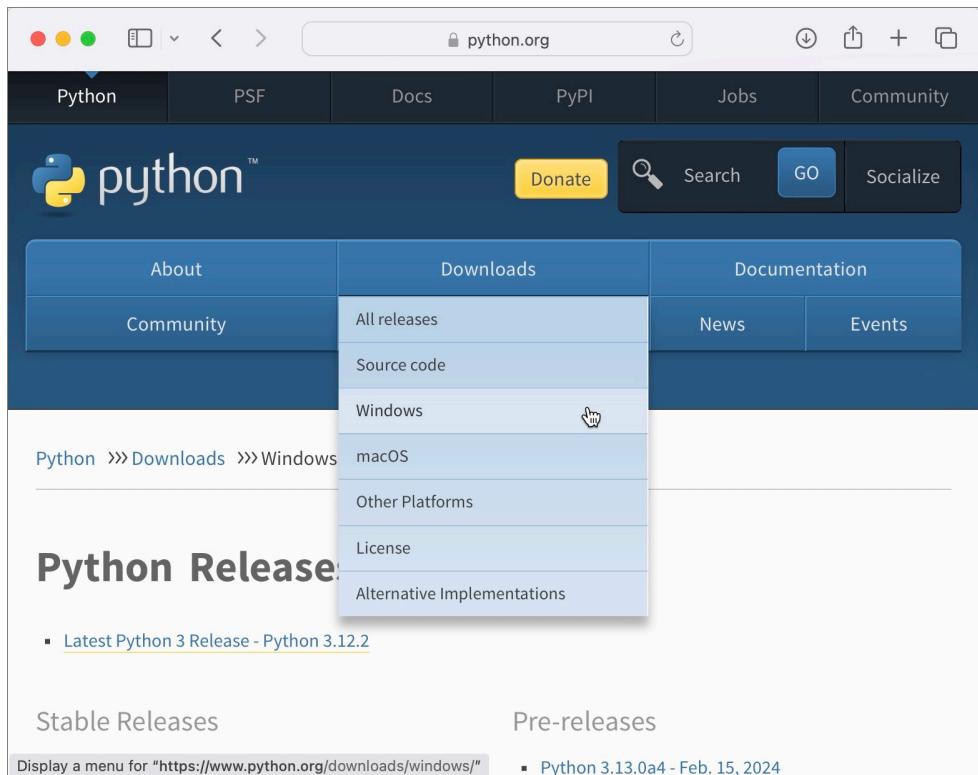
Visual Studio Code နှင့် Python အင်စတောလ်လုပ်ခြင်း

Visual Studio Code(VS Code) ဟာ ပရိုဂရမ်မာအများစုံ ကြိုက်နှစ်သက်တဲ့ မောဒန် ကုဒ်အယ်ဒီတာ တစ်ခုပါ။ Python, JavaScript, C++ စတဲ့ programming language အမျိုးမျိုးအတွက် အသုံးပြု နိုင်ပါတယ်။

PyCharm မှာတော့ ပရောဂျက်အသစ်ဆောက်ရင် Python ပါ တစ်ခါတည်း ဒေါင်းလုဒ်လုပ်ပြီး အင်စတောလ် လုပ်လို့ရတယ်။ VS Code နဲ့ Python ရေးမယ်ဆိုရင် Python Programming Language ကို သီး၌ ဒေါင်းလုဒ်လုပ်ပြီး အင်စတောလ် လုပ်ရပါမယ်။

Python အင်စတောလ်လုပ်ခြင်း

ဒီလုပ် <https://www.python.org/> ကိုဖွေ့စ်ပါ။ Download မိန္ဒီးမှ Windows နိုပ်ပါ (ပုံ က/၁၁ ကို ကြည့်ပါ)။ Apple ကွန်ပျူးတာအတွက် ဆိုရင် macOS ရွေးရပါမယ်။ လူများစုသုံးတဲ့ ပိုက်ခရီးဆော် ဝင်းခွီးအတွက် အင်စတောလ်လုပ်နည်းကို အခိုက်ပြေားမှာပါ။ ပုံ က/၁၂ မှာလို့ တွေ့ရပါလိမ့်မယ်။ အင်စတော်လာ ဗားရှင်း 3.12 ထဲက လက်ရှိအမြင့်ဆုံး ကိုရွေးပါ။ ဒီစာရေးချိန်မှာ 3.12.2 ဟာ အမြင့်ဆုံးဗားရှင်းပါ။ Windows Installer (64-bit) ကိုနိုပ်ပြီး ဒေါင်းလုဒ်လုပ်ပါ။



ပုံ က/၀၁

ဒေါင်းလုဒ်ပြီးရင် အင်စတော်လာဖိုင်ကို ညာကလစ်နိုပ်ပြီး Run as administrator လုပ်ပါ။ ပုံ (က/၁၃) ကိုကြည့်ပါ။ ပုံ (က/၁၄) မှာလို့ ဒိုင်ယာလော် ဆောက်စ် ပွင့်လာပါမယ်။ အနိဂုံးထားတဲ့ ချက်ချေ ဆောက်စ်နှစ်ခုကို ချက်ချေလုပ်ပြီး Install Now နိုပ်ပါ။ အင်စတောလ် ပြီးလို့ Setup was successful

Note that Python 3.12.2 cannot be used on Windows 7 or earlier.

- Download Windows embeddable package (32-bit)
- Download Windows embeddable package (64-bit)
- Download Windows embeddable package (ARM64)
- Download Windows installer (32-bit)
- Download Windows installer (64-bit)**
- Download Windows installer (ARM64)

▪ Python 3.12.1 - Dec. 8, 2023

Note that Python 3.12.1 cannot be used on Windows 7 or earlier.

- Download Windows embeddable package (32-bit)
- Download Windows embeddable package (64-bit)
- Download Windows embeddable package (ARM64)
- Download Windows installer (32-bit)
- Download Windows installer (64-bit)

▪ Python 3.13.0a2 - Nov. 21, 2023

- Download Windows embeddable package (32-bit)
- Download Windows embeddable package (64-bit)
- Download Windows embeddable package (ARM64)
- Download Windows installer (32-bit)
- Download Windows installer (64-bit)
- Download Windows installer (ARM64)

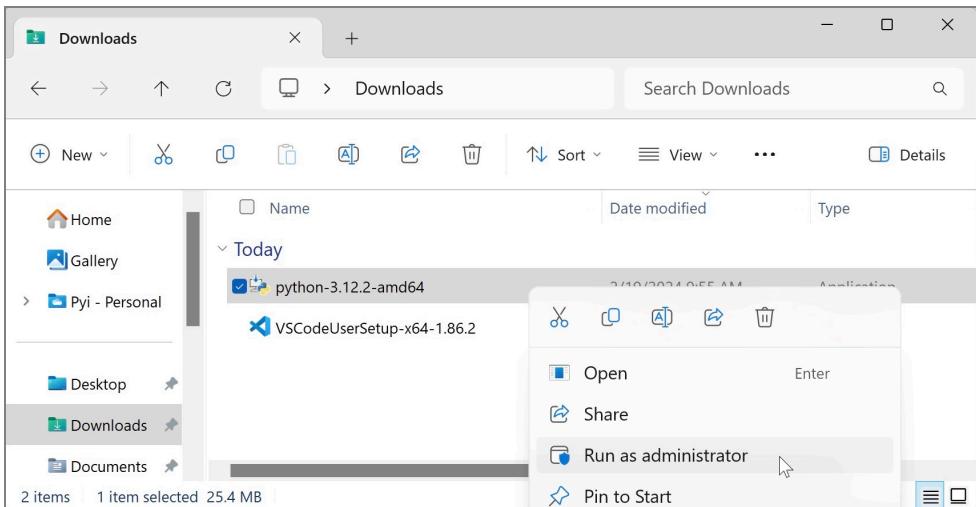
▪ Python 3.13.0a1 - Oct. 13, 2023

- Download Windows embeddable package (32-bit)
- Download Windows embeddable package (64-bit)
- Download Windows embeddable package (ARM64)
- Download Windows installer (32-bit)
- Download Windows installer (64-bit)
- Download Windows installer (ARM64)

ပုံ ၁၀၂

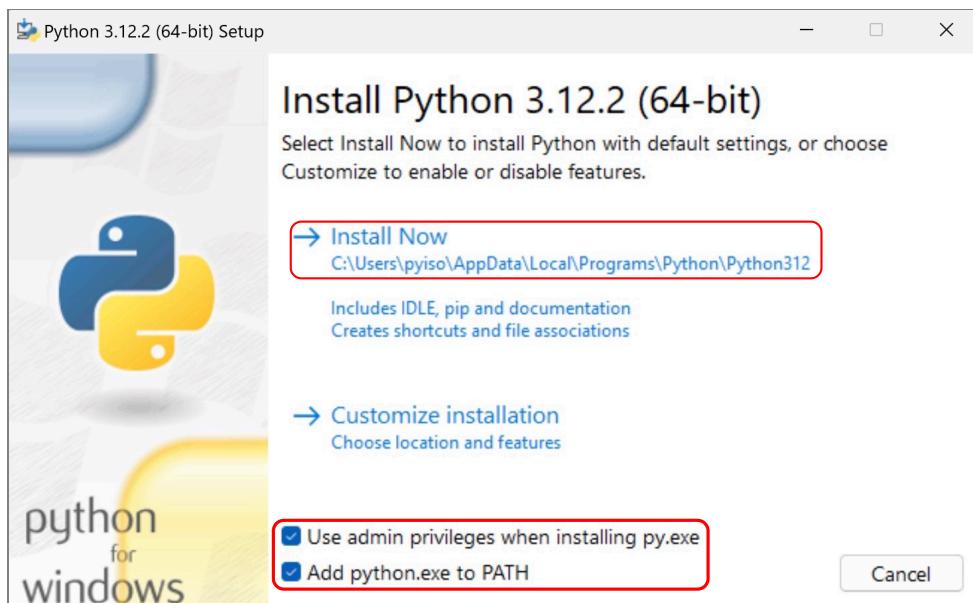
ပေါ်လာရင် Close ခလုတ်နိုပ်ပြီး ပိတ်ပါ။

ဝင်းဒီး command prompt (cmd) မှာ `python --version` run ရင် ပုံ (၁၁/၁၅) မှာလို အင် စတော်လုပ်ထားတဲ့ ဗားရှင်းကို ပြပေးသင့်ပါတယ်။



ပုံ ၁၀၃

၂၁



ပုံ ၃/၀၄

The screenshot shows a Microsoft Windows Command Prompt window. The title bar says "Command Prompt". The window displays the following text:
Microsoft Windows [Version 10.0.22621.2861]
(c) Microsoft Corporation. All rights reserved.
C:\Users\pyiso>python --version
Python 3.12.2
C:\Users\pyiso>

ပုံ ၃/၀၅

VS Code အင်စတေလုပ်ခြင်း

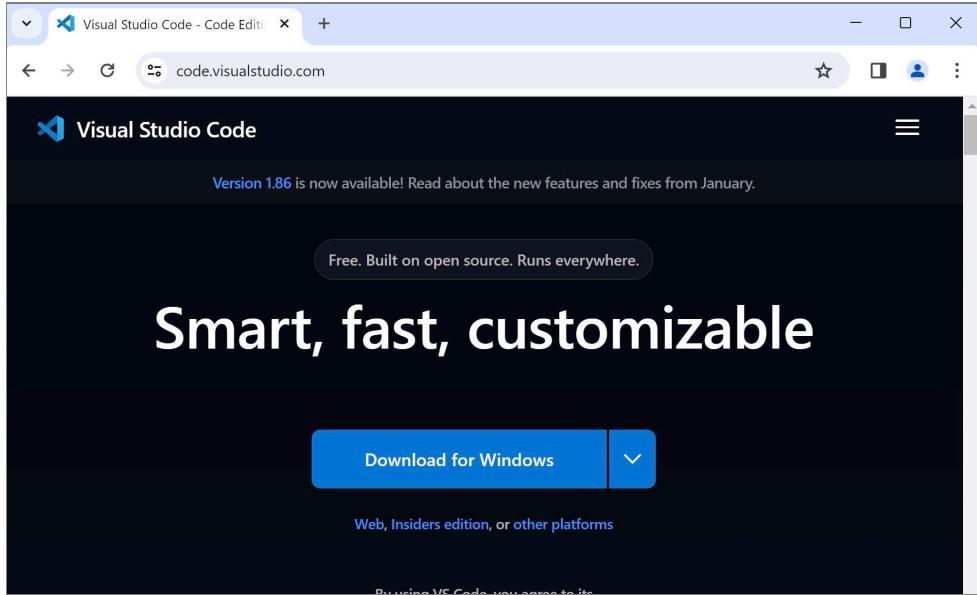
အခုနိရင် Python programming language အောင်မြင်စွာ ထည့်ပြီးသွားပါပြီ။ VS Code အင်စတေလုပ်ဆက်လုပ်ပါမယ်။ အင်စတေလုပ်ဆက်လုပ်ရန် ဝဘ်စာမျက်နှာကို အောက်ပါလုပ်လုပ်ပါ။

<https://code.visualstudio.com>

မှတစ်ဆင့် သွားပါ။ ပုံ (က/၁၆) ဝဘ်စာမျက်နှာကို တွေ့ရပါမယ်။ [Download for Windows](#) နှိပ်၍ ဒေါင်းလုပ်လုပ်ပါ။

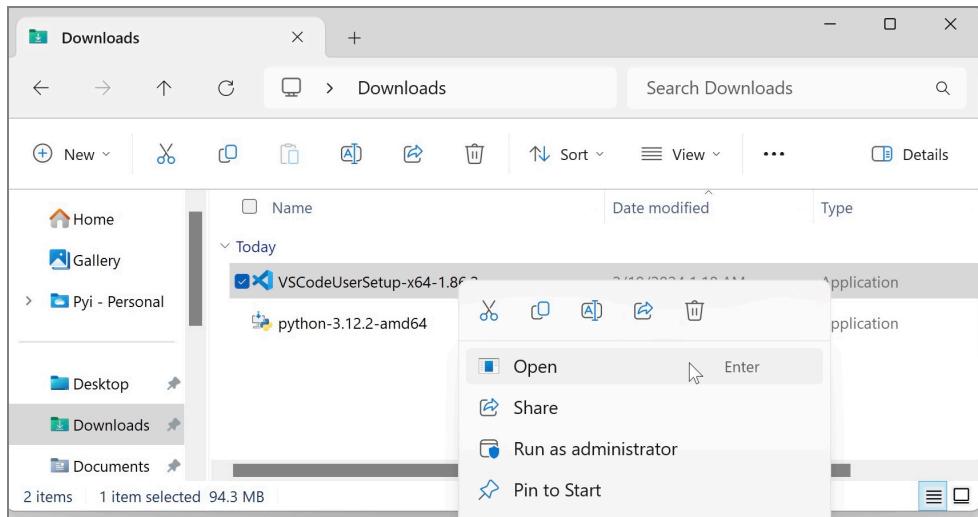
ပြီးတဲ့အခါ အင်စတေလုပ်ကို ညာကလစ်နိုင်ပြီး ဖွင့်ပါ (ပုံ က/၁၇ မှာ ပြထားပါတယ်)။ အင်စတေလုပ် ဒိုင်ယာလျှော်ဘောက်စ် ပွင့်လာရင် [I accept the agreement](#) ကို ချက်ချွဲလုပ်၍ Next > တစ်ခု ပြီးတစ်ခု ဆက်နှိပ်သွားပြီး နောက်ဆုံးမှာ Install နှိပ်ပါ။ အင်စတေလုပ်နေတာကို ခက္ခဘာင့်ပြီး၊ ပြီး သွားရင် Finish နှိပ်ပါ။

Welcome စခရင်ကို ပုံ (က/၁၈) လို တွေ့ရပါမယ်။ [Dark Modern](#) (သို့) [Light Modern](#) နှစ်သက်ရာ သို့မဟုတ် ရွေးပါ။ ဒါဆိုရင် VS Code လည်း အင်စတေလုပ်လုပ်ပါပြီ။ ကားရဲ့လုပ်ရုံးမှာ ရေးဖို့ ဆက်လုပ်ရပါမယ်။

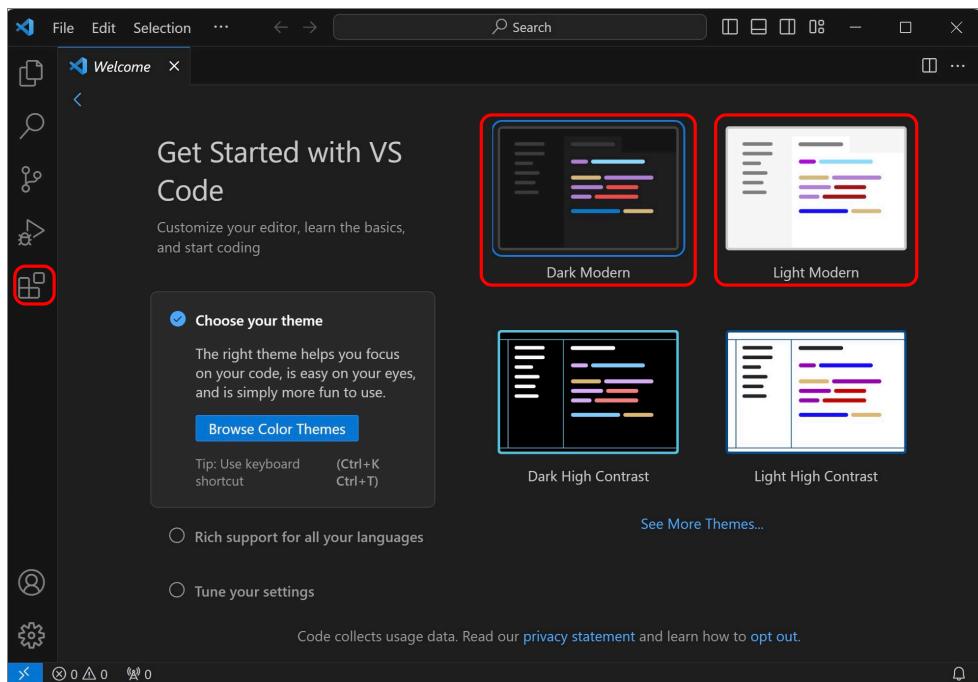


ပုံ က/၁၆

J2



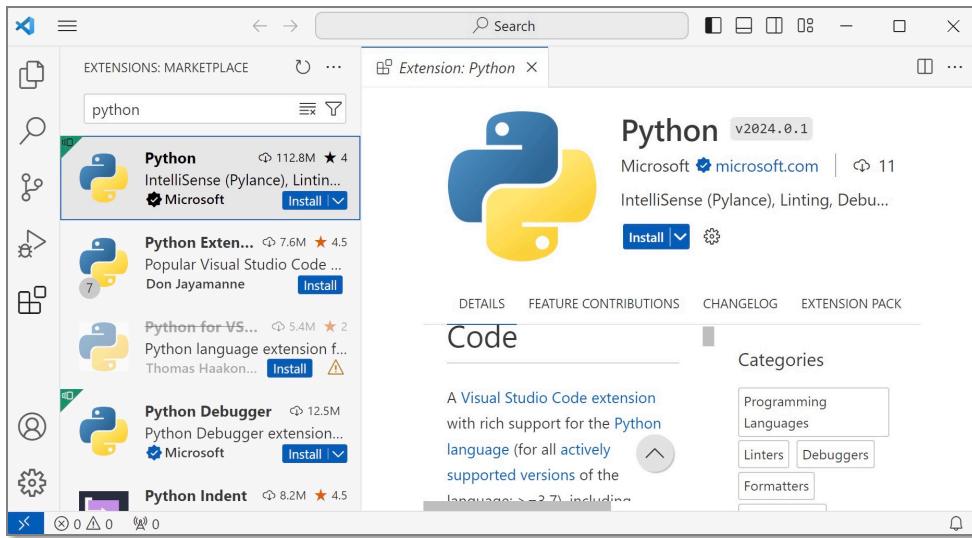
ଓঁ গো



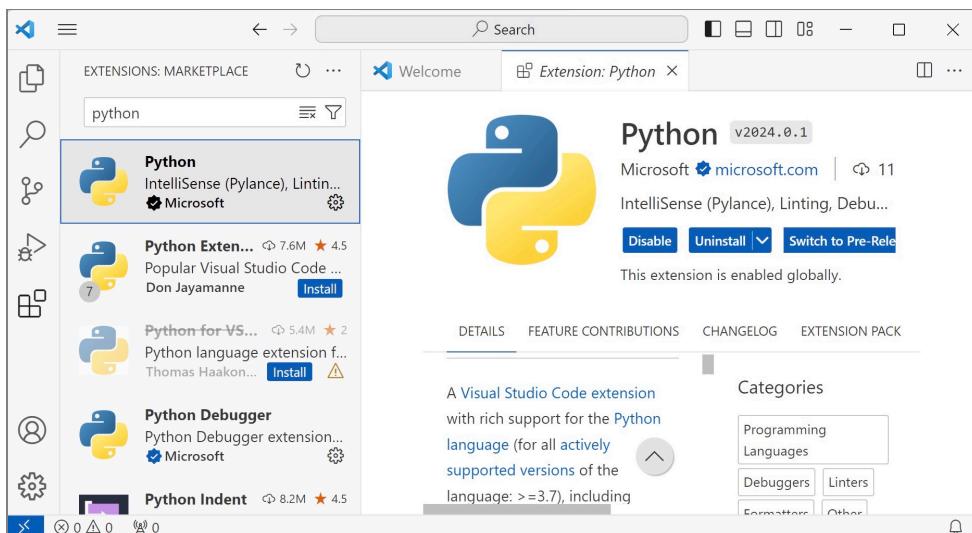
ଓঁ গো

VS Code Python Extension တည်ခြင်း

Python ပရိဂုဂ္ဂများဖို့အတွက် VS Code က ဒီအတိုင်းဆိုရင် သိပ်အဆင်မပြေသေးပါဘူး။ Python အတွက် extension အင်စတောလ် လုပ်ပေးရပါအိုးမယ်။ ပုံ (က/၁၈) ဘယ်ဘက်သော်နားမှာ အနီးပိုင်းထားတဲ့ အိုင်ကွန်လေးကို နှိပ်ပါ။ Python extension ကိုရှာပါ။ ပုံ (က/၁၉) မှာ ဘယ်လိုရှာရမလဲ ပြထားပါတယ်။ ပုံမှာတွေ့ရတဲ့ Microsoft က ထုတ်တဲ့ extension ကို အင်စတောလ်လုပ်ပါ။ အောင်မြင်ရင် ပုံ (က/၂၀) မှာလို ဖြစ်သွားပါမယ်။ Python နဲ့ VS Code ကိစ္စတော့ ပြီးသွားပြီ။ ကားရဲ့လဲ example run ဖို့ ဆက်လပ်ရမယ်။



ပုံ က/၁၉



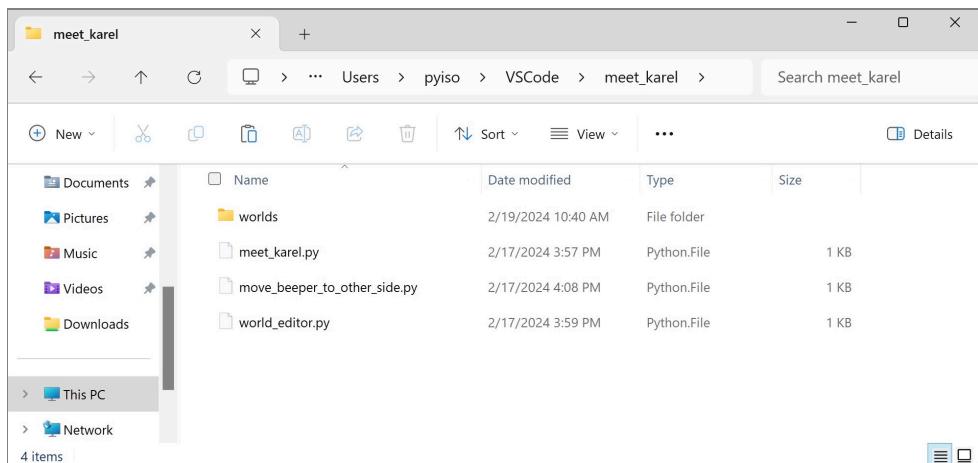
ပုံ က/၂၀

နမူနာ ကားရဲလ် ကဗ္ဗာနှင့် ပရိဂရမ်ဘ် ဖိုင်များထည့်ခြင်း

meet_karel.zip ဖိုင်ကို ဒီလင့် <http://tinyurl.com/3mm9c7j> ကနေ အောင်လုပ်လုပ်ပါ။ ငှါး zip ဖိုင်ကို extract လုပ်ပါ။ meet_karel နံမည်နဲ့ ဖိုဒါတစ်ခု ရလာပါမယ်။ ဖိုဒါထဲ ဝင်ကြည့်ရင် အောက်ပါ အတိုင်း ရှိသင့်ပါတယ်။

■ worlds

- ▶ meet_karel.w
- ▶ move_beeper_to_other_side.w
- meet_karel.py
- move_beeper_to_other_side.py
- world_editor.py



ပုံ က/ဂ

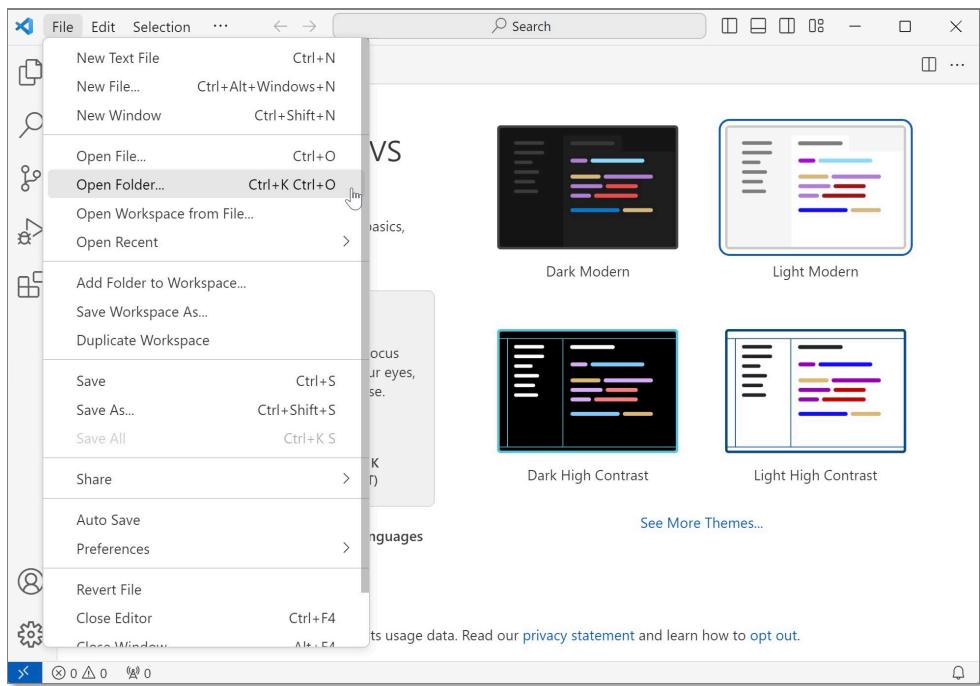
VS Code အတွက် ဖိုဒါတစ်ခုကို မိမိအတွက် အဆင်ပြေမဲ့နေရာမှာ သီးသန့်တည်ဆောက် ထားသင့်တယ်။ ဥပမာ

C:\Users\yourname\VS Code

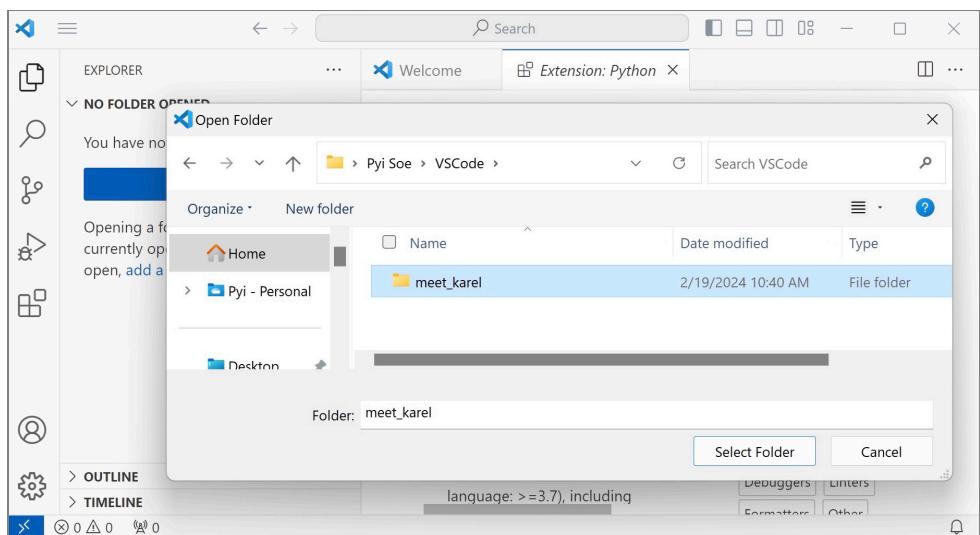
မိမိ လက်ရှိ Home ဖိုဒါကို C: drive ရဲ့ Users ဖိုဒါထဲမှာ တွေ့နိုင်ပါတယ်။ Win + R ရေးကတ် နှိပ်ပြီး %userprofile% ရှိက်ထည့်။ Ok လုပ်ပြီး Home ဖိုဒါကို သွားနိုင်ပါတယ်။ အကယ်၍ မသွားတတ်ရင်လည်း ပြဿနာမရှိပါဘူး။ ကိုယ့်အတွက် လွယ်ကူမဲ့ Desktop, Downloads, Documents တစ်ခုခုထဲမှာ VS Code အတွက် ဖိုဒါတစ်ခု ထားလည်းရတယ်။

`meet_karel` ဖိုဒါ (.zip ဖိုင်မဟုတ်ပါ) ကို အထက်ပါအတိုင်း အသစ် ဆောက်ထားတဲ့ VS Code သီးသန့်ဖိုဒါထဲကို ကော်ပီကူးထည့်ပါ။ ငှါး `meet_karel` ဖိုဒါကို VS Code `File` မိုးမှ `Open Folder` နှိပ်၍ ဖွေ့စွဲပါ။ ပုံ (က/ဂ) တွင်ကြည့်ပါ။

အခန်းအလိုက် နမူနာ ကုဒ်ဖိုင်တွေ ထည့်ပေးထားတဲ့ .zip ဖိုင်တွေကိုလည်း အထက်ပါအတိုင်း လုပ်ရပါမယ်။ .zip ဖိုင်ကို ဖြော် ရလာတဲ့ ဖိုဒါကို သီးသန့်ဖိုဒါတစ်ခုထဲကို ကော်ပီကူးထည့်၊ VS Code နဲ့ အဲ ဒီဖိုဒါကို ဖွေ့စွဲပါပဲ။



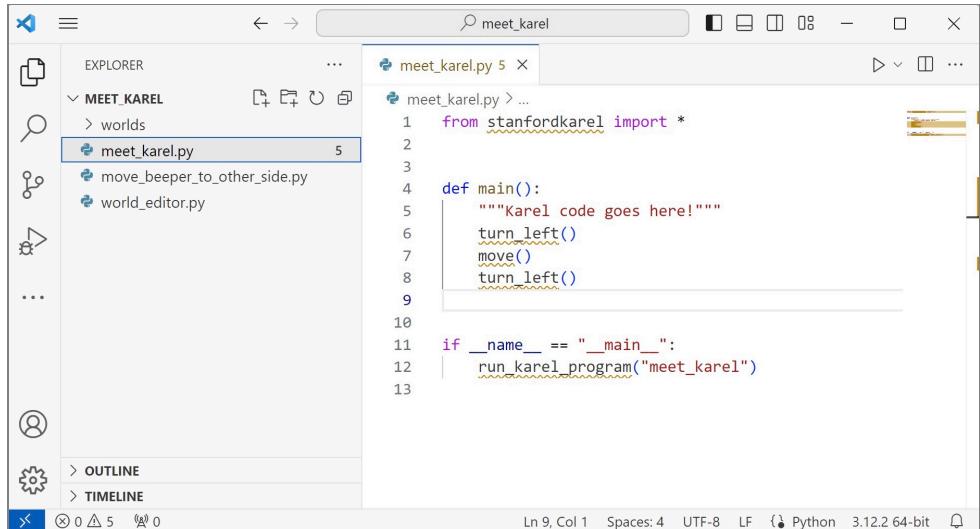
ပုံ က/၂၂



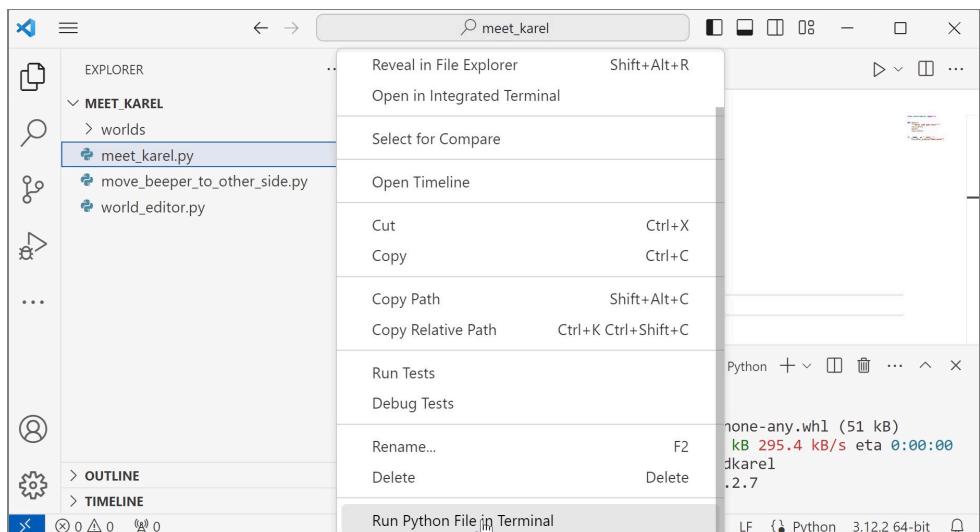
ပုံ က/၂၃

stanfordkarel လိုက်ဘရီ အင်စတောလ်လုပ်ခြင်း

`meet_karel.py` ဖိုင်ကို ကလစ်နှစ်ချက်နှင့် ဖွင့်ပါ။ ကုဒ်အယ်ဒီတာ ပွင့်လာမယ် (ပုံ ၂/၂၅)။ အဲဒီ ကုဒ် အယ်ဒီတာပေါ် (သို့) `meet_karel.py` ဖိုင်ကို ညာကလစ်နှင့်ပြီး `Run Python File in Terminal` လုပ်ပါ (ပုံ ၂/၂၅)။ Terminal ပွင့်လာပြီး အယ်ရာမက်ဆောင်တွေ ပြလိမ့်မယ်။ ပုံ (၂/၂၆) မှာ ကြည့်ပါ။ ကဗျားရုံးပရှုဂျုပ်အတွက် လိုအပ်တဲ့ stanfordkarel လိုက်ဘရီ အင်စတောလ် မလုပ်ရသေးပါဘူး။ ဒါ ကြောင့် အယ်ရာဖြစ်နေတာ။



ပုံ ၂/၂၆



ပုံ ၂/၂၇

ခုနကပွင့်လာတဲ့ Terminal မှာပဲ အောက်ပါကွန်မန်းကို run ပြီး stanfordkarel လိုက်ဘရီကို အင်စတောလ်လုပ်ပါ။

The screenshot shows the VS Code interface. The Explorer sidebar on the left lists files in a folder named 'MEET_KAREL', including 'worlds', 'meet_karel.py', 'move_beeper_to_other_side.py', and 'world_editor.py'. The 'meet_karel.py' file is selected and open in the main editor area. The code in the editor is:

```
from stanfordkarel import *
def main():
    pass
```

Below the editor is a terminal window showing the command PS C:\Users\pyiso\VSCode\meet_karel> & C:/Users/pyiso/AppData/Local/Programs/Python/Python312/python.exe c:/Users/pyiso/VSCode/meet_karel/meet_karel.py and its output:

```
PS C:\Users\pyiso\VSCode\meet_karel> & C:/Users\pyiso/AppData/Local/Programs/Python/Python312/python.exe c:/Users/pyiso/VSCode/meet_karel/meet_karel.py
Traceback (most recent call last):
  File "c:/Users/pyiso/VSCode/meet_karel/meet_karel.py", line 1,
    in <module>
      from stanfordkarel import *
ModuleNotFoundError: No module named 'stanfordkarel'
PS C:\Users\pyiso\VSCode\meet_karel> pip install stanfordkarel
```

The status bar at the bottom indicates the file is at Ln 9, Col 1, with 5 spaces, using UTF-8 encoding, and is a Python 3.12.2 64-bit file.

ပုံ ၃/၂၆

`pip install stanfordkarel`

ပုံ (၃/၂၆) မှာ အနိဂင်းထားတာကို ကြည့်ပါ။ အဲဒီအတိုင်းရိုက်ထည့်ပြီး Enter ကိုနိပ်ပါ။ ခဏကြောတဲ့ အခါ အခုလို မက်ဆွဲချုပ်တွေ ကျလာပါလိမ့်မယ်။

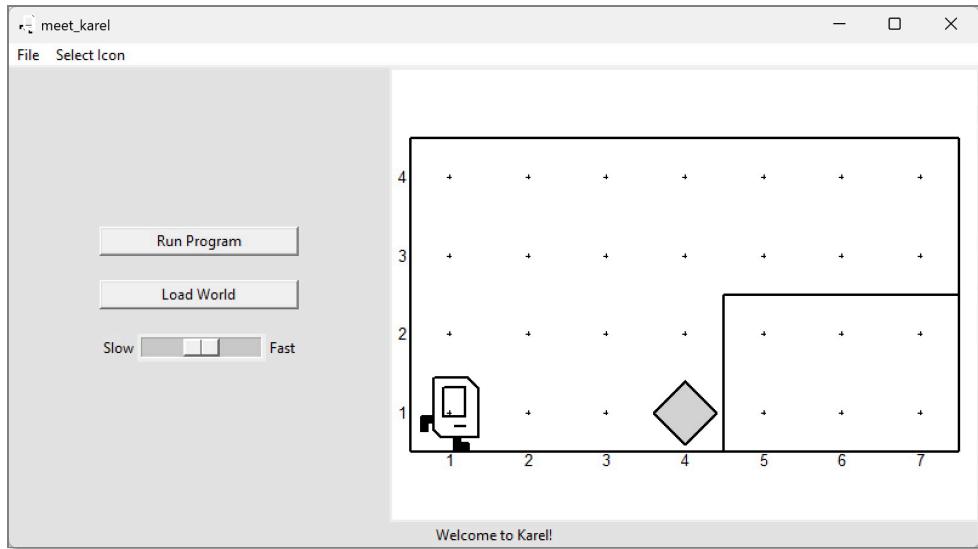
```
ModuleNotFoundError: No module named 'stanfordkarel'
PS C:\Users\pyiso\VSCode\meet_karel> pip install stanfordkarel
Collecting stanfordkarel
  Downloading stanfordkarel-0.2.7-py3-none-any.whl (51 kB)
           ━━━━━━━━━━━━ 51.9/51.9 kB 295.4 kB/s eta 0:00:00
Installing collected packages: stanfordkarel
Successfully installed stanfordkarel-0.2.7
PS C:\Users\pyiso\VSCode\meet_karel>
```

ဟိုကလိုက်ပြထားတဲ့ မက်ဆွဲချုပ် တွေရရင် အင်စတောလ် အောင်မြင်လိုပါ။ ပုံ (၃/၂၄) အယ်ဒီတာမှာလို သတိပေး လိုင်းတွေနဲ့မျဉ်းတွေ မရှိသင့်တော့ဘူး။ stanfordkarel လိုက်ဘရဲ့ အင်စတောလ် လုပ်ပြီးပြီးလို့ သတိပေးတာတွေ ပျောက်သွားသင့်တယ်။

meet_karel.py ဖိုင်ကို ညာကလစ်နှုပ်ပြီး Run Python File in Terminal ပြန်လုပ်ကြည့်ပါ။ ပုံ (၃/၂၇) မှာပြထားတဲ့ ကားရဲ့ပရိုကရမ် ပွင့်လာသင့်ပါတယ်။ Run Program နှုပ်ကြည့်ပါ။ စက်ရှုပ် လေး ကားရဲ့ နေရာရွှေသွားတာ တွေရမယ်။ meet_karel.py ကို အောက်ပါအတိုင်း ဖြည့်စက်ရေးပါ။

```
from stanfordkarel import *

def main():
    """Karel code goes here!"""
    move()
    move()
    move()
    pick_beeper()
```



ဦး က/ဂျ

```

turn_left()
move()
move()

turn_left()
turn_left()
turn_left()

move()
put_beeper()

if __name__ == "__main__":
    run_karel_program("meet_karel")

```

`meet_karel.py` ဖိုင်ကို ညာကလစ်နိုပ်ပြီး Run Python File in Terminal ပြန်လုပ်ကြည့်ပါ။ ကား ရဲလ်ပရိုကရမ် ပွင့်လာရင် Run Program နှိပ်ကြည့်ပါ။ ဘိပါလေးကို နေရာရွှေ့ပေးပါလိမ့်မယ်။ အကယ်၍ ကားရဲလ်ပရိုကရမ် မတက်လာရင် ကုဒ်ရေးတာမှားနေလို့ ဖြစ်နိုင်တယ်။ အပေါ်ကုဒ်နဲ့ နှိုင်းယှဉ်ပြီး ကြည့်ပါ။ VS Code အယ်ဒီတာမှာ အနိုင်းတွေ့လေးတွေ ပြတဲ့နေရင် အဲဒီနေရာတွေမှာ ဆင်းတက်စုံမှားနေတာ ဖြစ်နိုင်တယ်။

VS Code အယ်ဒီတာမှာ ပရိုကရမ်ကုဒ်ပြင်ပြီး ပြန် run တဲ့အခါ ပထမ run ထားတဲ့ ပရိုကရမ် ကို အရင်ပိတ်ဖိုလိုပါတယ်။ ဆိုလိုတာက `meet_karel.py` ကို run ထားတယ်ဆိုပါစို့။ ပုံ (က/ဂျ) က ဝင်းဒီးပွင့်လာမယ်။ `meet_karel.py` ကုဒ်ကို ပြင်ပြီး ပြန် run ချင်ရင် အဲဒီ ဝင်းဒီးကို အရင်ပိတ်ရ မယ်။ မဟုတ်ရင် ပြင်ထားတဲ့ ပရိုကရမ်က ချက်ချင်း ပွင့်မလာဘူး။ ပထမ ဟာကို ပိတ်တော့မှုပဲ နောက် run တဲ့ဟာ ပွင့်လာမှာ။

ဂိုက်ကွင်းကျွန်နေတာက အဖြစ်များတဲ့ အမှားပါ။ ကျွန်ခဲ့လို့ မရပါဘူး။ အင်ဒန်တေးရှင်း (indentation)

tion) လုပ်ရမဲ့နေရာမှာ မလုပ်ထားရင်လည်း ပြဿနာဖြစ်တယ်။ move, turn_left တွေကို ဘေးမျဉ်းလှာဘက်ဆွဲပြီး အင်ဒန်လုပ်ပေးရမယ်။ အဒါတွေ ဂရာမစိုက်မိရင် ဆင်းတက်စံအမှားဖြစ်ပြီး ပရိုကရမ် run လို့ မရနိုင်ဘူး။

Terminal မှာ ထုတ်ပေးတဲ့ မက်ဆွဲချုပ်တွေကို ကြည့်ပြီးတော့လည်း ဘာပြဿနာဖြစ်နေလဲ မှန်းဆလို့ရနိုင်တယ်။ ဘာကြောင့်ဖြစ်နိုင်လဲ ဆက်စပ်စဉ်းစားလို့ ရတယ်။ ဥပမာ ဖြစ်တဲ့ပြဿနာအလိုက် အခုလို တွေ့ရပါမယ်။

```
File "c:\Users\pyiso\VS Code\meet_karel\meet_karel.py", line 6
    move(
        ^
SyntaxError: '(' was never closed

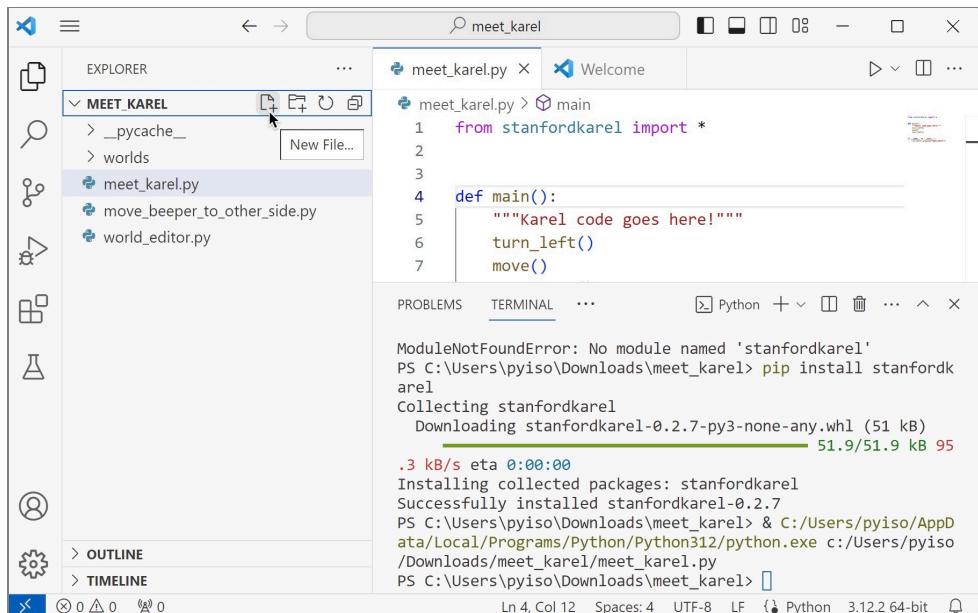
File "c:\Users\pyiso\VS Code\meet_karel\meet_karel.py", line 7
    move()
        ^
IndentationError: unexpected indent

Traceback (most recent call last):
  File "c:\Users\pyiso\VS Code\meet_karel\meet_karel.py", line 1,
    in <module>
      from stanfordkarel import *
ModuleNotFoundError: No module named 'stanfordkarel'
```

VS Code တွင် Python ဖိုင် အသစ်ယူခြင်း

MEET_KAREL ပင်မ ပရောဂျက်ဖိုဒ်အပေါ်မှာ ကလစ်နိုင်ပါ။ ပုံမှာ ပြထားတဲ့ New File အင်ကွန်ကိုနှိပ်ပါ။ ဖိုင်နံပည်ဖြည့်တဲ့ ဘောက်စံလေး ပေါ်လာမယ်။ Python ဖိုင်တွေက .py အိပ်စံတန်းရှင်းနဲ့ ဖြစ်ရပါမယ်။ ဒါကြောင့် နံပည် ဖြည့်တဲ့အခါ .py နဲ့ အဆုံးသတ်ပေးရပါမယ် (ဥပမာ hello.py)။ ကားချုပ်ပရီဂရမ်တစ်ခုကို Python ဖိုင်တစ်ခု ထားပါမယ်။ ပင်မ ပရောဂျက်ဖိုဒ်အပေါ်မှာပဲ တိုက်ရှိကြရပါမယ်။

နောက်ပိုင်း၊ အဆင့်မြင့်လာရင် ပရီဂရမ်တစ်ခုအတွက် ပရောဂျက်တစ်ခု ထားနိုင်တယ်။ ကုဒ်ဖိုင်တွေ အပြင် ပရီဂရမ်အတွက် လိုအပ်တဲ့ ရုပ်ပုံတော့၊ အခြားဖိုင်တွေ (config ဖိုင်၊ setting ဖိုင် စသည်ဖြင့်) လည်း ပါနိုင်တယ်။ ပင်မပရောဂျက် အောက်မှာပဲ ဖိုင်တွေက တိုက်ရှိကြရဖို့လည်း မလိုတော့ဘူး။ ဆက်စပ်ရာ ဖိုင်တွေကို အမျိုးအစားအလိုက်၊ ဖန်ရှင်အလိုက် ဖို့ဒါတွေခဲ့ပြီး စနစ်ကျ စီစဉ်ဖွဲ့စည်း ထားရမှာပါ။ ပရောဂျက်တစ်ခုမှာ ဖိုင်တွေကို စနစ်တကျ စွဲထားဖို့ အရေးကြီးပါတယ်။



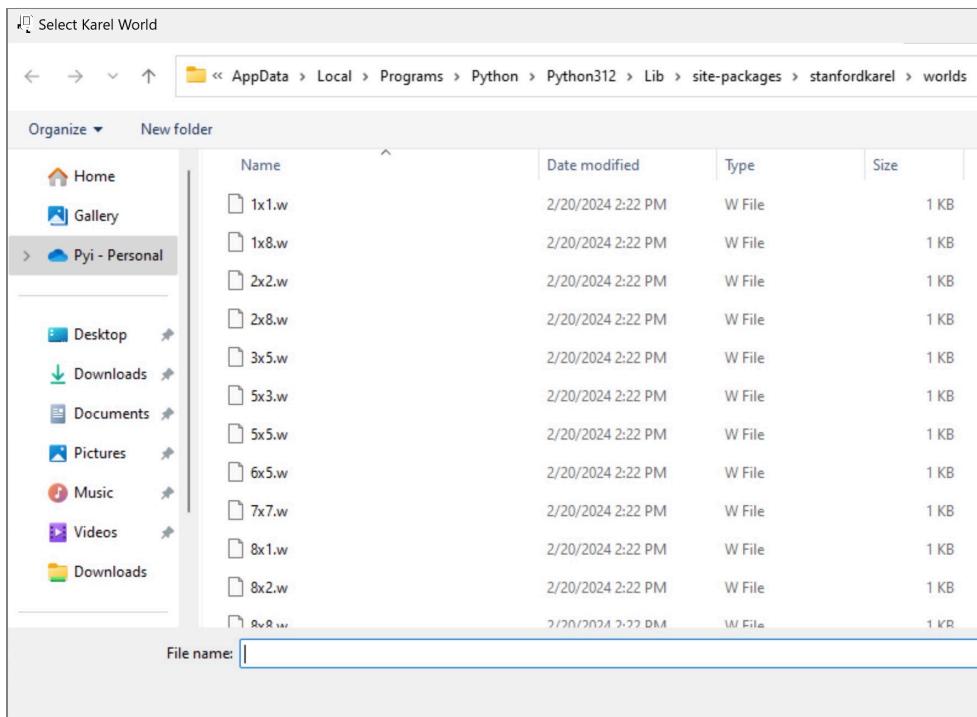
ပုံ က/ဂ

နောက်ဆက်တဲ့ ခ

ကားရဲလ်ပရီဂရမ် ဖီချာများ

ကားရဲလ် ကဗ္ဗာဖိုင်များ

ကားရဲလ်ပရီဂရမ် ဝင်းခွဲးမှာ **Load World** ခလုတ်နှုပ်ပြီး ကဗ္ဗာဖိုင်အသစ် တင်လိုရတယ်။ ခလုတ်နှုပ်လိုက် ရင် အခုလို ဖိုင် ခိုင်ယာလော့ဂုံးပွင့်လာမယ်။



ပုံ ၉/၁

ဒါက stanfordkarel လိုက်ဘရဲ သူနိုင်အရိုးအတိုင်း ပါတဲ့ worlds ဖို့ပါ။ ဖိုင်တွေက .w နဲ့ ဆုံးပါတယ်။ ကဗ္ဗာဖိုင်တွေကို ပင်မ ပရောဂျက်အောက် worlds ဖို့ပါထဲမှာ ထားလိုလည်းရတယ်။ အခြားနေရာတွေမှာ ထားလိုတော့ မရဘူး။

စာအုပ်ပါ ဥပမာတွေ၊ လေ့ကျင့်ခန်းတွေအတွက် ကဗ္ဗာဖိုင်တွေကို ပရောဂျက် တစ်ခုချင်းအလိုက် သီးခြား worlds ဖိုဒါထဲမှာ ထည့်ပေးထားမှာပါ။ ပရိုဂ်ရမ်တစ်ခုဟာ ကဗ္ဗာတစ်ခုတည်းမှာပဲ အလုပ်လုပ်တာမဟုတ်ဘဲ အလားတူ ကဗ္ဗာအမျိုးမျိုးအတွက် အလုပ်လုပ်အောင် ရေးပေးရတာ။ အခုပြောတာကို နား မလည်ရင် အခန်း (၂) ဖတ်ပြီးရင် နားလည်သွားမှာပါ။

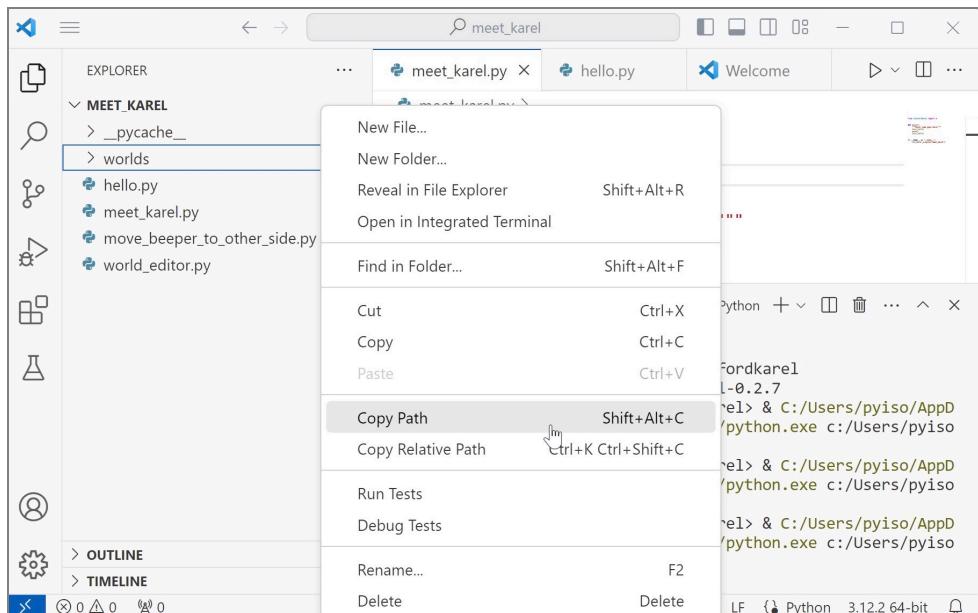
Load World လုပ်တဲ့အခါ ပွင့်လာတဲ့ ဖိုင် ခိုင်ယာလော်ကာ ကိုယ်လိုချင်တဲ့ worlds ဖိုဒါ မဖြစ် နေဘူး။ သူနိုင်ပါတဲ့ worlds ဖိုဒါ ဖြစ်နေတယ်။ ကိုယ်ခေါ်တင်ချင်တဲ့ ဖိုင်တွေရှိတဲ့ လက်ရှိပရောဂျက်ရဲ့ worlds ဖိုဒါကို သွားရမယ်။ ဥပမာ PyCharm/VS Code အတွက် MeetKarel/meet_karel ပရောဂျက် worlds ဖိုဒါ လမ်းကြောင်း အပြည့်အစုံက

C:\Users\yourname\VSCode\meet_karel\worlds

C:\Users\yourname\PycharmProjects\MeetKarel\worlds

ဖြစ်ပါမယ်။ ဖိုင်ခိုင်ယာလော်ကနေ ဖော်ပြပါ လက်ရှိပရောဂျက် worlds ဖိုဒါကို တစ်ဆင့်ချင်း သွားပြီး တင်ချင်တဲ့ ကဗ္ဗာဖိုင် (.w ဖိုင်) ကို ရွေးရမှာပါ။

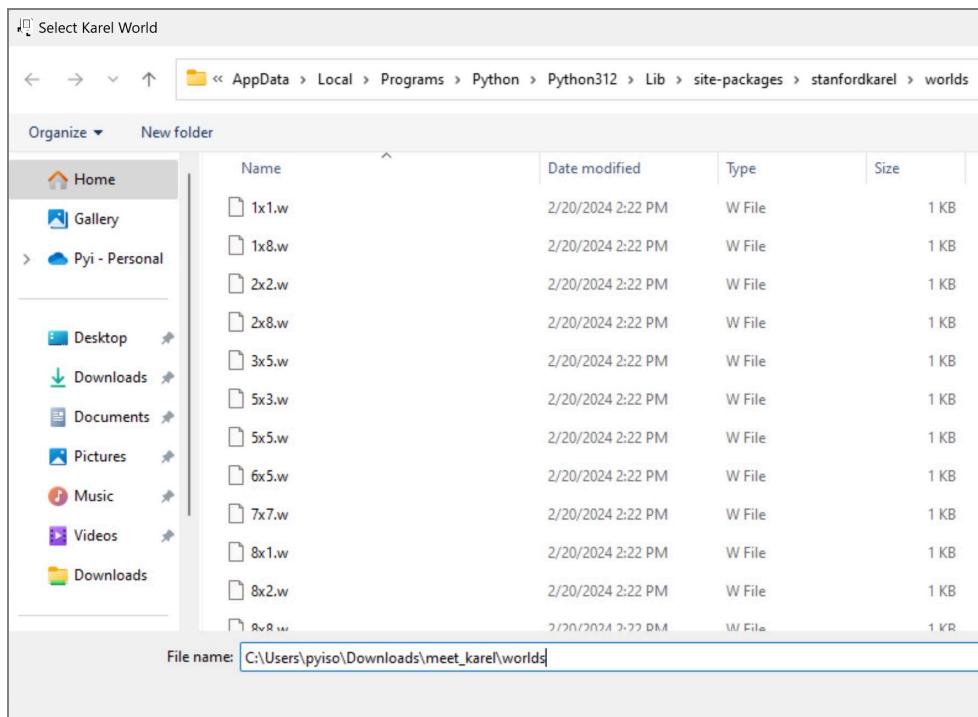
အပေါ်ကနည်းနဲ့ အဆင်မပြောရင် အခုလုပ်စမ်းကြည့်ပါ။ လက်ရှိပရောဂျက် worlds ဖိုဒါကို ညာ ကလစ်နှုပ်ပြီး Copy Path လုပ်ပါ (ပုံ ၁/၂)။ ဖိုင် ခိုင်ယာလော် **File name** မှာ ကူးထည့်ပါ (ပုံ ၂/၂)။ **Enter** ကိုနှိပ်ပါ။ ပရောဂျက် worlds ဖိုဒါကိုရောက်သွားပါမယ်။ လိုတဲ့ကဗ္ဗာဖိုင် ရွေးတင်ရုံပါပဲ။ ပုံ (၁/၂) မှာ meet_karel worlds ကို နူးနှုပ်ထားပါတယ်။



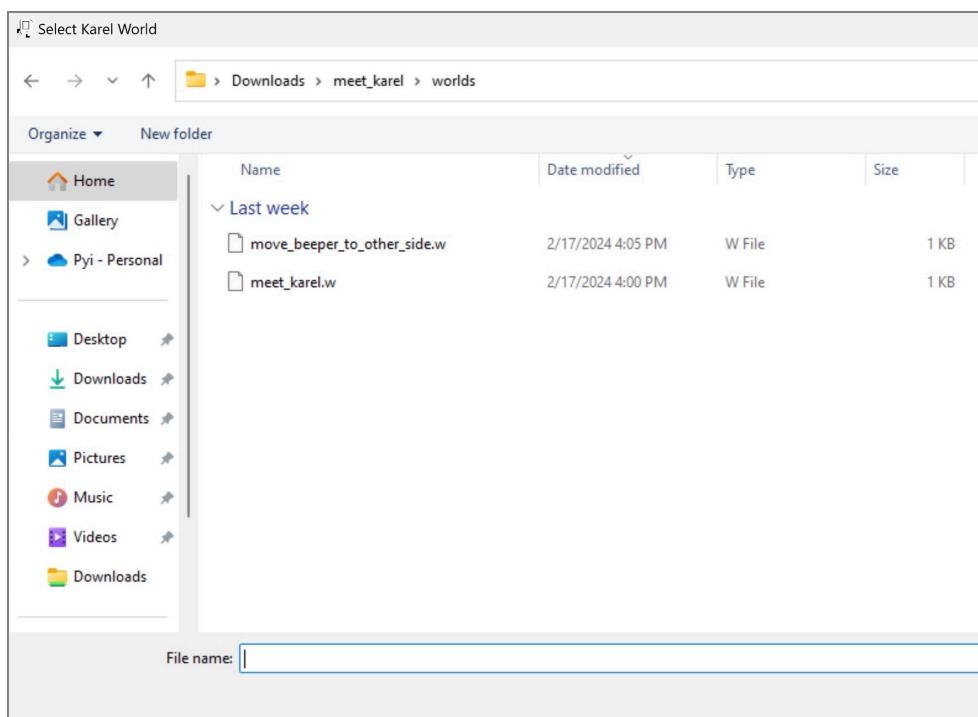
ပုံ ၁/၂

အကယ်၍ အထက်ဖော်ပြပါနည်းတွေက ရှုပ်နေတယ်ထင်ရင် မှတ်ရ/သွားရ လွယ်တဲ့ Desktop, Downloads, Documents လို နေရာတစ်ခုခုမှာ သီးသန့်ဖိုဒါတာစ်ခု အောက်ပြီး ပရောဂျက်အားလုံး ထည့်ထားတာ အရှင်းဆုံးပါပဲ။ ပရောဂျက်ဖိုဒါနေရာ သိရင် ဖိုင်ခိုင်ယာလော်ကနေ ဘယ်လိုမဆို ရောက်အောင် သွားလိုရပါတယ်။

၃။



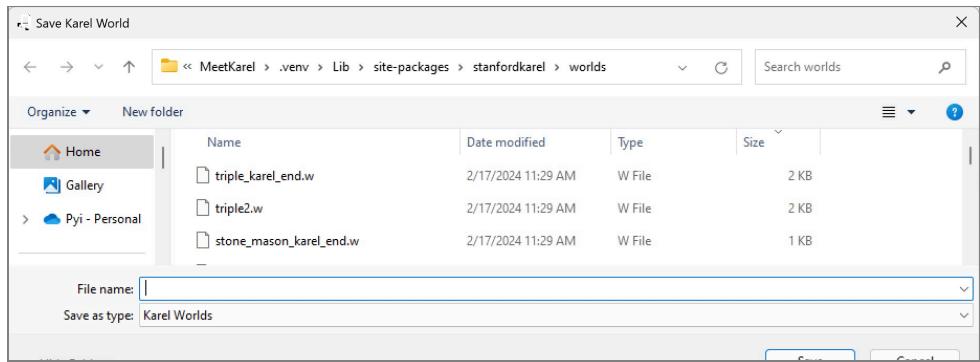
ပုံ ၅/၃



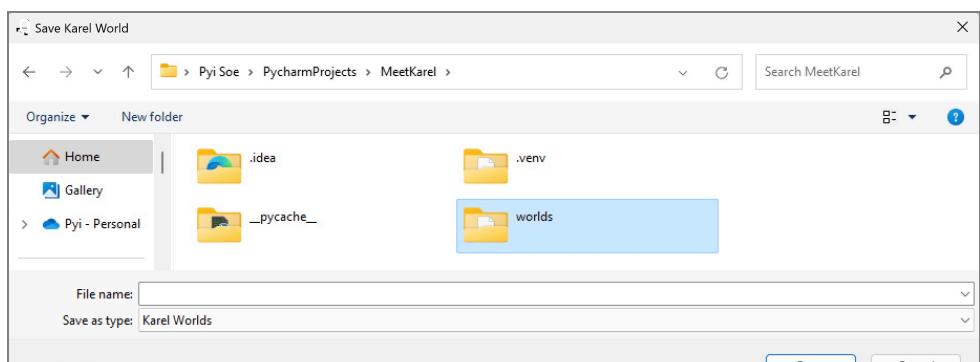
ပုံ ၆/၄

ကိုယ်ပိုင် ကားရဲလ်ကမ္ဘာ ဆောက်ခြင်း

ကားရဲလ်ရဲ ကမ္ဘာ အသစ်တစ်ခုဆောက်မယ်ဆိုရင် world_editor.py ဖိုင်ကို ညာကလစ်နှင့် Run ပါ။ Would you like to load an existing world? လို ပေါ်လာပြီး Yes/No ရွှေ့ခိုင်ပါလိမ့်မယ်။ No ကို နှိပ်ပါ။ ကမ္ဘာအရွယ်အစားအတွက် ကော်လံ ဘယ်နှစ်ခုလဲ row ဘယ်နှစ်ခုလဲ ထည့်ပေးပါ။ ကိုယ်ပိုင် ကားရဲလ်ကမ္ဘာ တည်ဆောက်လိုရတဲ့ ဝင်းဒီးပွင့်လာပါလိမ့်မယ်။ ကားရဲလ် မျက်နှာမူရာအပ်၊ ဘိပါအိတ်ထဲရှိ ဘိပါအရေအတွက်၊ နံရုံဆောက်/ဖျက်တာ၊ ဘိပါထည့်/ဖယ်ထုတ်တာ စတာတွေ လုပ်နိုင်ပါတယ်။ Save World နှိပ်ပြီး သိမ်းနိုင်ပါတယ် ဖိုင်ကိုသိမ်းတဲ့အခါ သုန္တရှိ သိမ်းခိုင်းတဲ့ဖို့ (default world folder) ထဲမှာ သို့မဟုတ် ပင်မ ပရောဂျက်ဖို့တဲ့က worlds ဖို့ဒါထဲမှာ သိမ်းရပါမယ်။



ပုံ ၉/၅



ပုံ ၉/၆

