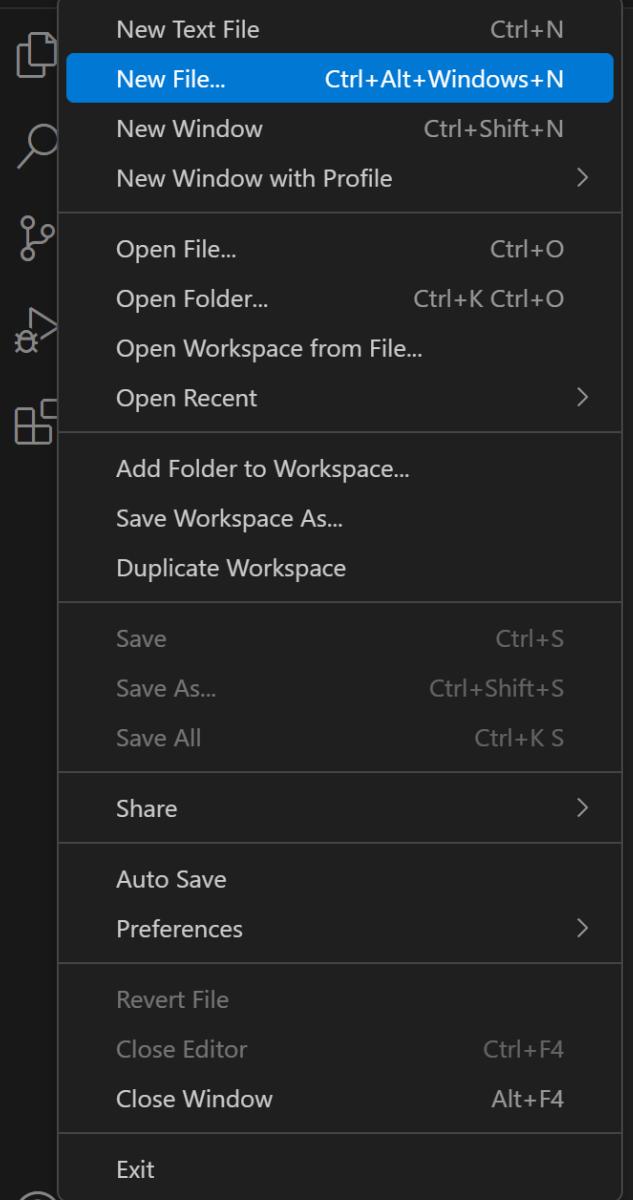


2D Game (2)

2025-09-24



Show All Commands `Ctrl + Shift + P`

Open File `Ctrl + O`

Open Folder `Ctrl + K Ctrl + O`

Open Recent `Ctrl + R`

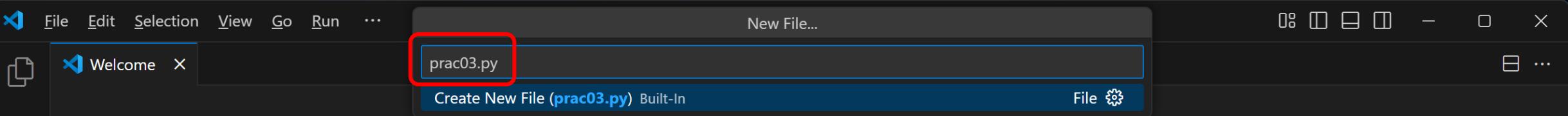
Open Chat `Ctrl + Alt + I`



Restricted Mode

0 ▲ 0





Visual Studio Code

Editing evolved

Start

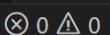
- New File...
- Open File...
- Open Folder...
- Connect to...
- Generate New Workspace...

Recent

- ARGallery C:\Users\sunje\UnityProjects
 - Pong C:\Users\sunje\UnityProjects
 - task0408 C:\Users\sunje\Desktop\SM_unity
 - task_ing_after C:\Users\sunje\Desktop\SM_unity
 - ShaderTest C:\Users\sunje\UnityProjects
- [More...](#)

Walkthroughs

- Get started with VS Code**
Customize your editor, learn the basics, and start coding
- Get Started with Python Development** New
- Learn the Fundamentals**



← → ▼ ↑



▶ 바탕 화면 > Commit

▼ C

Commit 검색



구성

새 폴더



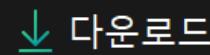
홈



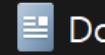
갤러리



▶ 바탕 화면



▼ 다운로드



Documents



파일 이름(N): prac03.py



파일 형식(I): All Files (*.*)



▲ 폴더 숨기기

“저장”하기 원하는
폴더 위치 지정

prac01.py

prac02.py



school.png



Create File

취소



Welcome

prac03.py X

D ▾ E ...



C: > Users > sunje > Desktop > Commit > prac03.py



⊗ 0

△ 0

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.13.7

Enemies appeared!



SlimeA



SlimeB



SlimeC



Hero



Fight

Run

게임 제작 (1)

1. 화면 구성 및 캐릭터 이동

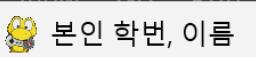
- **화면 구성** – 게임 창 설정 후 게임 화면의 크기와 배경색을 설정
- **캐릭터 생성 및 이동** – 간단한 캐릭터를 화면에 표시하고, 키보드 입력을 받아 캐릭터를 상하좌우로 이동

2. 맵 생성 및 충돌 처리

3. 적 생성 및 전투 시스템

4. 아이템 생성 및 게임 완성

```
prac03.py ●  
C: > Users > sunje > Desktop > Commit > prac03.py > ...  
1 # Created by 윤종현  
2  
3 import pygame  
4  
5 # -----  
6 # 게임 초기화  
7 # -----  
8 pygame.init() # pygame 라이브러리 초기화  
9  
10 screen_width, screen_height = 800, 600 # 화면 크기 설정  
11 screen = pygame.display.set_mode((screen_width, screen_height)) # 화면 생성  
12 pygame.display.set_caption("본인 학번, 이름")  
13 clock = pygame.time.Clock() # 게임 속도 조절용 시계  
14  
15 # -----  
16 # 게임 루프  
17 # -----  
18 running = True  
19 while running:  
20     for event in pygame.event.get():  
21         if event.type == pygame.QUIT: # 종료 버튼 누르면 게임 종료  
22             running = False  
23  
24     # -----  
25     # 화면 그리기  
26     # -----  
27     screen.fill((40,60,90)) # 배경색  
28  
29     pygame.display.flip() # 화면 업데이트  
30     clock.tick(60) # 초당 60프레임  
31  
32     pygame.quit() # 게임 종료  
33
```



본인 학번, 이름



 1 prac01.py  prac02.py  prac03.py

```
C: > Users > sunje > Desktop > Commit > prac03.py > ...
1 # Created by 윤종현
2
3 import pygame
4
5 # -----
6 # 게임 초기화
7 # -----
8 pygame.init() # pygame 라이브러리 초기화
9
10 screen_width, screen_height = 800, 600 # 화면 크기 설정
11 screen = pygame.display.set_mode((screen_width, screen_height)) # 화면 생성
12 pygame.display.set_caption("본인 학번, 이름") # 게임 속도 조절용 시계
13 clock = pygame.time.Clock() # 글꼴 설정 (None = 기본 글꼴, 28 = 글자 크기)
14 font = pygame.font.Font(None, 28) # 자주 쓰는 색상 미리 정의
15
16 colors = { # 자주 쓰는 색상 미리 정의
17     "WHITE": (255,255,255), "RED": (200,0,0), "BLUE": (50,100,200),
18     "GREEN": (0,200,0), "YELLOW": (200,200,0)
19 }
20
21 # -----
22 # 화면에 글자 그리는 함수
23 # -----
24 def draw_text(screen, font, text, pos, color, center=False):
25     # 글자를 화면에 렌더링
26     image = font.render(text, True, color)
27     # 글자 위치 지정
28     rectangle = image.get_rect(center=pos) if center else image.get_rect(topleft=pos)
29     # 화면에 붙이기
30     screen.blit(image, rectangle)
31
32 # -----
33 # 체력바 그리는 함수
34 # -----
```

File Edit Selection View Go Run ... ← → Search

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
31
32 # -----
33 # 체력바 그리는 함수
34 # -----
35 def draw_hp(screen, x, y, width, height, hp, maxhp, colors):
36     # 체력바 배경 (흰색 테두리)
37     pygame.draw.rect(screen, colors["WHITE"], (x-2, y-2, width+4, height+4))
38     # 체력바 기본 빨간색
39     pygame.draw.rect(screen, colors["RED"], (x, y, width, height))
40     # 체력바 현재 체력만큼 녹색으로 표시
41     pygame.draw.rect(screen, colors["GREEN"], (x, y, int(width*max(0,hp/maxhp)), height))
42
43 # -----
44 # 캐릭터/몬스터 클래스
45 # -----
46 class Creature:
47     def __init__(self, name, hp):
48         self.name = name          # 이름
49         self.max = hp             # 최대 체력
50         self.hp = hp              # 현재 체력
51
52 # -----
53 # 게임 실행
54 # -----
55 # 플레이어 캐릭터 생성
56 player = Creature("Hero", 50)
57
58 # -----
59 # 게임 루프
60 # -----
61 running = True
62 while running:
63     for event in pygame.event.get():
64         if event.type == pygame.QUIT:      # 종료 버튼 누르면 게임 종료
```

Ln 57, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.13.7



prac01.py

prac02.py

prac03.py ●

C: > Users > sunje > Desktop > Commit > prac03.py > ...



```
51
52     # -----
53     # 게임 실행
54     #
55     # 플레이어 캐릭터 생성
56     player = Creature("Hero", 50)
57
58     # -----
59     # 게임 루프
60     #
61     running = True
62     while running:
63         for event in pygame.event.get():
64             if event.type == pygame.QUIT:      # 종료 버튼 누르면 게임 종료
65                 running = False
66
67         # -----
68         # 화면 그리기
69         #
70         screen.fill((40,60,90))          # 배경색
71
72         # 플레이어 그리기
73         pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
74         draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
75         draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
76
77         pygame.display.flip()            # 화면 업데이트
78         clock.tick(60)                 # 초당 60프레임
79
80     pygame.quit()                   # 게임 종료
81
```



⊗ 0 △ 0

Ln 76, Col 1 Spaces: 4 UTF-8 CRLF {} Python 📈 3.13.7 🔍



Hero

게임 제작 (2)

1. 화면 구성 및 캐릭터 이동

2. 맵 생성 및 충돌 처리

3. 적 생성 및 전투 시스템

- 적 캐릭터 생성 – 간단한 AI를 가진 적 캐릭터를 생성하고 맵에 배치
- 전투 시스템 구현 – 플레이어와 적 간의 기본적인 전투 메커니즘을 구현 (예: 공격력, 체력 등)

4. 아이템 생성 및 게임 완성

File Edit Selection View Go Run ... ← → Search

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
42
43     # -----
44     # 캐릭터/몬스터 클래스
45     # -----
46     class Creature:
47         def __init__(self, name, hp):
48             self.name = name          # 이름
49             self.max = hp            # 최대 체력
50             self.hp = hp            # 현재 체력
51
52     # -----
53     # 게임 실행
54     # -----
55     # 플레이어 캐릭터 생성
56     player = Creature("Hero", 50)
57
58     # 적 캐릭터 생성
59     enemies = [
60         Creature("SlimeA", 30),
61         Creature("SlimeB", 25),
62         Creature("SlimeC", 20)
63     ]
64
65     # -----
66     # 게임 루프
67     # -----
68     running = True
69     while running:
70         for event in pygame.event.get():
71             if event.type == pygame.QUIT:      # 종료 버튼 누르면 게임 종료
72                 running = False
73
74     # -----
75     # 화면 그리기
```

x ⌂ 0 ▲ 0 Ln 64, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.13.7

File Edit Selection View Go Run ... ← →

prac01.py prac02.py prac03.py x

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```

64
65  # -----
66  # 게임 루프
67  # -----
68  running = True
69  while running:
70      for event in pygame.event.get():
71          if event.type == pygame.QUIT:      # 종료 버튼 누르면 게임 종료
72              running = False
73
74      # -----
75      # 화면 그리기
76      # -----
77      screen.fill((40,60,90))           # 배경색
78
79      # 적 그리기
80      gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
81      for i, enemy in enumerate(enemies):
82          x, y = left + i*gap, 120
83          pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
84          draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
85          draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
86
87      # 플레이어 그리기
88      pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
89      draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
90      draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
91
92      pygame.display.flip()             # 화면 업데이트
93      clock.tick(60)                  # 초당 60프레임
94
95  pygame.quit()                   # 게임 종료

```

$\text{screen_width} // 2$
 $y = 120 \rightarrow$
 $\text{gap} = 150 \quad \text{gap} = 150 \quad w = 80$
 \uparrow
 $x = \text{screen_width} // 2 - ((3-1)*\text{gap} + w) // 2$
 $= (\text{screen_width} - (3-1)*\text{gap} + w) // 2$

※ enemy들을 가로 정중앙에 배치하는 대신
 $w // 2$ 만큼 오른쪽으로 이동시켰음
(계산량도 줄일 수 있음)

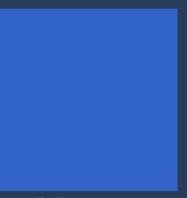
Ln 86, Col 1 Spaces: 4 UTF-8 CRLF {} Python 3.13.7



SlimeA

SlimeB

SlimeC



Hero

연습 문제 (1)

- **마우스** 클릭으로 메뉴 선택하기

File Edit Selection View Go Run ... 🔍 Search

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
65  # -----
66  # 게임 루프
67  #
68  options = ["Fight", "Run"]
69  choice = 0
70  running = True
71  while running:
72      for event in pygame.event.get():
73          if event.type == pygame.QUIT:
74              running = False
75
76  # -----
77  # 화면 그리기
78  #
79  screen.fill((40,60,90))           # 배경색
80
81  # 적 그리기
82  gap, left = 150, (screen_width - (len(enemies)-1) * 150) /
83  for i, enemy in enumerate(enemies):
84      x, y = left + i*gap, 120
85      pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
86      draw_text(screen, font, enemy.name, (x+40, y+90), color
87      draw_hp(screen , x, y-20, 80, 10, enemy.hp, enemy.max,
88
89  # 플레이어 그리기
90  pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
91  draw_text(screen, font, player.name, (140,390), colors["WH
92  draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, c
93
94  # 메뉴 옵션 표시
95  for i, option in enumerate(options):
96      draw_text(screen, font, option, (screen_width//2, 400+i*40), (0,200,200) if i==choice else colors["WHITE"], True)
97
98  pygame.display.flip()             # 화면 업데이트
```

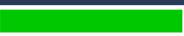
← screen_width//2 →

y = 400 ➡ Fight
= 40
Run

Ln 96, Col 94 Spaces: 4 UTF-8 CRLF {} Python 3.13.7



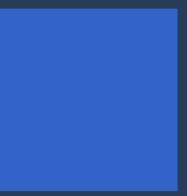
SlimeA



SlimeB



SlimeC



Hero

Fight

Run

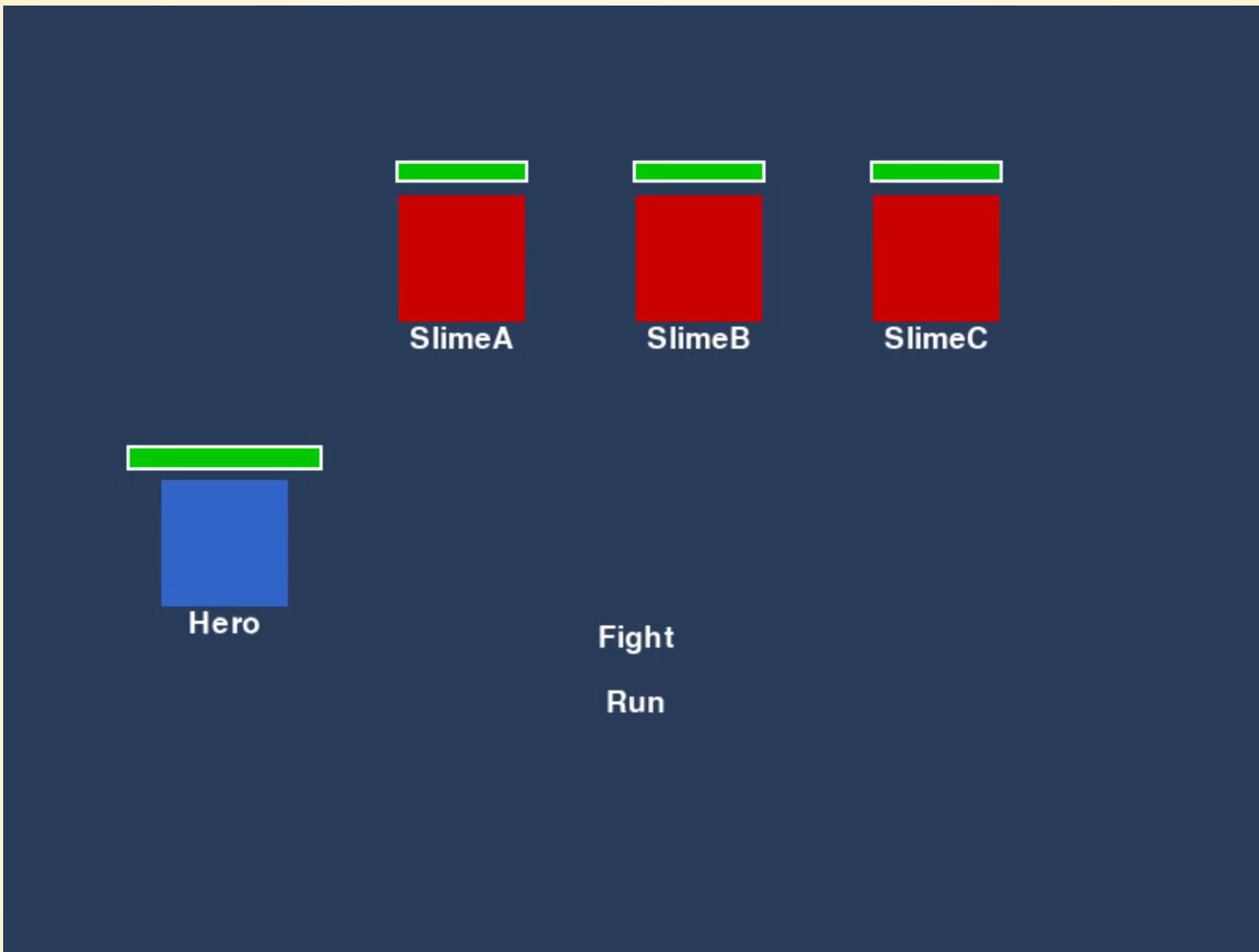
The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, and a search bar. The left sidebar features icons for file operations like Open, Save, Find, and Preferences. The main workspace displays a Python script named `prac03.py`. The code implements a game loop using Pygame, handling mouse input, drawing enemies and player, and displaying a menu. A specific line of code at the bottom of the loop is highlighted with a red border.

```
C: > Users > sunje > Desktop > Commit > prac03.py > ...  
77     mouse_pos = pygame.mouse.get_pos()      # 마우스의 위치를 받아서 mouse_pos에 저장  
78     for i, option in enumerate(options):  
79         button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)  
80         if mouse_pos[0] > button.x and mouse_pos[0] < button.x+button.width:  
81             if mouse_pos[1] > button.y and mouse_pos[1] < button.y+button.height:  
82                 choice = i  
83  
84     # -----  
85     # 화면 그리기  
86     # -----  
87     screen.fill((40,60,90))                # 배경색  
88  
89     # 적 그리기  
90     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2  
91     for i, enemy in enumerate(enemies):  
92         x, y = left + i*gap, 120  
93         pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))  
94         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)  
95         draw_hp(screen , x, y-20, 80, 10, enemy.hp, enemy.max, colors)  
96  
97     # 플레이어 그리기  
98     pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))  
99     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)  
100    draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)  
101  
102    # 메뉴 옵션 표시  
103    for i, option in enumerate(options):  
104        draw_text(screen, font, option, [screen_width//2, 400+i*btn_gap], (0,200,200) if i==choice else colors["WHITE"], True)  
105  
106    pygame.display.flip()                  # 화면 업데이트  
107    clock.tick(60)                      # 초당 60프레임  
108  
109    pygame.quit()                      # 게임 종료  
110
```

Line 104 is highlighted with a red border:

```
draw_text(screen, font, option, [screen_width//2, 400+i*btn_gap], (0,200,200) if i==choice else colors["WHITE"], True)
```

Bottom status bar: Ln 104, Col 72 Spaces: 4 UTF-8 CRLF {} Python 3.13.7



연습 문제 (2)

- 전투 시스템 구현하기
 - 플레이어 – 스킬 보유
 - “Fight” 메뉴 선택하면, 플레이어가 보유한 스킬 보여주기

The screenshot shows a Python code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** Search
- Toolbar:** Includes icons for file operations like Open, Save, Find, and Run.
- Code Area:** Displays the file `prac03.py` with the following content:

```
43     # -----
44     # 공격 스킬 클래스
45     #
46     class Skill:
47         def __init__(self, name, power, hit):
48             self.name = name    # 기술 이름
49             self.power = power # 공격력 배율
50             self.hit = hit      # 명중률 (%)
51
52     # -----
53     # 캐릭터/몬스터 클래스
54     #
55     class Creature:
56         def __init__(self, name, hp, attack, skills):
57             self.name = name      # 이름
58             self.max = hp        # 최대 체력
59             self.hp = hp         # 현재 체력
60             self.attack = attack # 공격력
61             self.skills = skills # 가지고 있는 기술 리스트
62
63     # -----
64     # 게임 실행
65     #
66     # 플레이어 캐릭터 생성
67     player = Creature("Hero", 50, 5, [Skill("Slash", 4, 85), Skill("Fire", 6, 75)])
68
69     # 적 캐릭터 생성
70     enemies = [
71         Creature("SlimeA", 30, 3, [Skill("Bite", 3, 80)]),
72         Creature("SlimeB", 25, 3, [Skill("Bite", 3, 85)]),
73         Creature("SlimeC", 20, 2, [Skill("Bite", 2, 90)])
74     ]
75
76     # -----
```
- Right Panel:** Shows a vertical stack of tabs and code snippets, likely from a code repository or history.
- Status Bar:** Ln 46, Col 12, Spaces: 4, UTF-8, CRLF, {}, Python, 3.13.7

File Edit Selection View Go Run ... ⏪ ⏩ Search

C: > Users > sunje > Desktop > Commit > prac03.py > [e] level

```
76  # -----
77  # 게임 루프
78  #
79  options = ["Fight", "Run"]          # 메뉴 버튼 글자
80  choice, level = -1, 0             # 선택한 메뉴 (choice=0 첫번째 메뉴, choice=2 두번째 메뉴, ...), 메뉴 레벨
81  message = "Enemies have appeared!" # 메시지 출력
82  btn_width, btn_height, btn_gap = 60, 30, 40      # 메뉴 버튼 크기, 간격 (변경 가능)
83  running = True
84  while running:
85      for event in pygame.event.get():
86          if event.type == pygame.QUIT:           # 종료 버튼 누르면 게임 종료
87              running = False
88          if event.type == pygame.MOUSEBUTTONDOWN: # 마우스 버튼이 눌렸을 때
89              mouse_pos = pygame.mouse.get_pos()    # 마우스의 포지션을 받아서 mouse_pos에 저장
90
91          # -----#
92          # 메인 메뉴 선택 (Fight / Run)
93          # -----
94          if level == 0:
95              for i, option in enumerate(options):
96                  button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
97                  if mouse_pos[0] > button.x and mouse_pos[0] < button.x+button.width:
98                      if mouse_pos[1] > button.y and mouse_pos[1] < button.y+button.height:
99                          choice = i
100                         if choice == 0:           # Fight 메뉴 선택
101                             level, choice = 1, -1
102                         elif choice == 1:         # Run 메뉴 선택
103                             message = "You can't run away!"
104
105                         # -----
106                         # 스킬 선택
107                         #
108                         elif level == 1:
109                             for i, skill in enumerate(player.skills):
button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
```

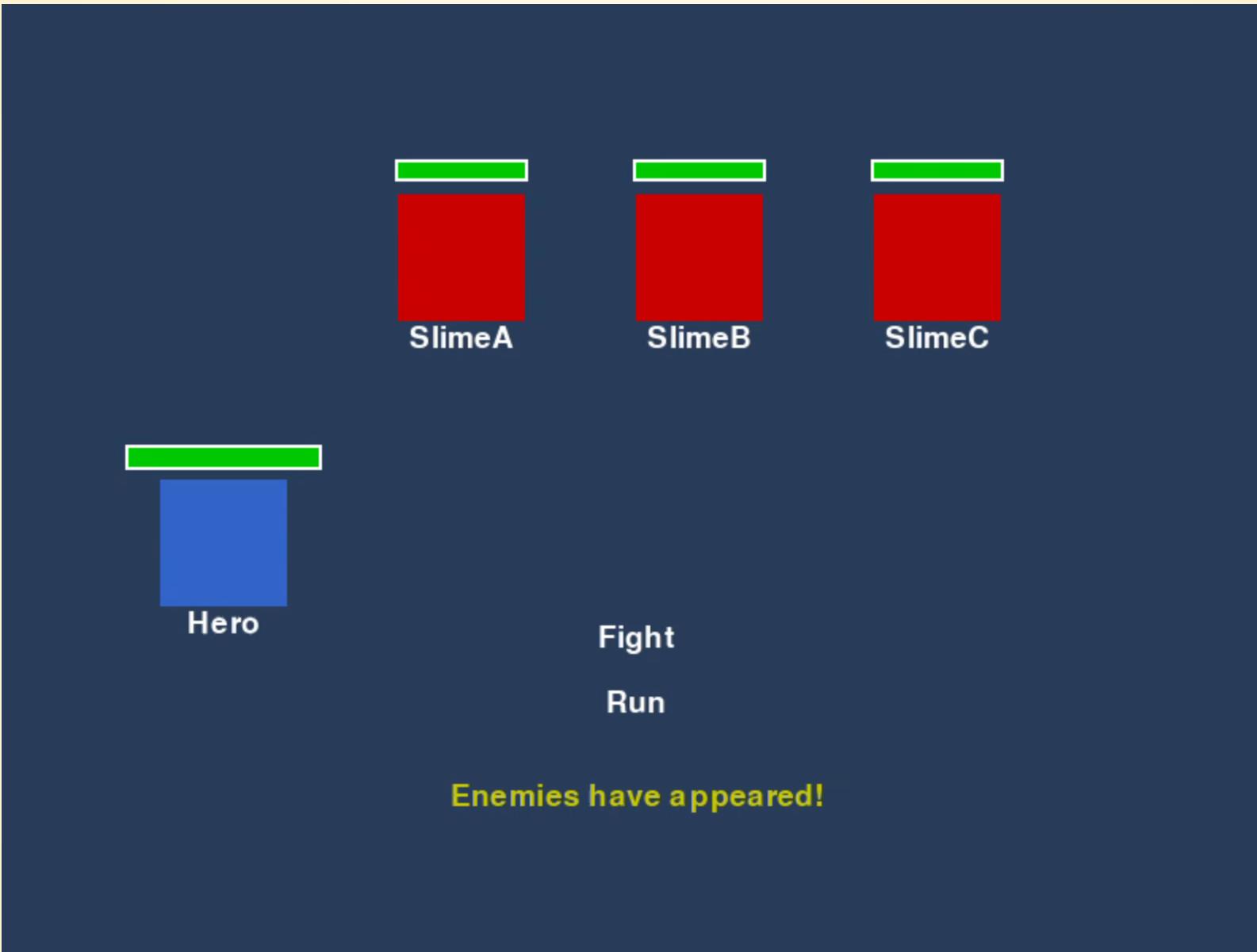
Ln 80, Col 14 Spaces: 4 UTF-8 CRLF {} Python 3.13.7

File Edit Selection View Go Run ... ⏪ ⏩ Search

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
104     # -----
105     # 스킬 선택
106     # -----
107     elif level == 1:
108         for i, skill in enumerate(player.skills):
109             button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
110             if mouse_pos[0] > button.x and mouse_pos[0] < button.x+button.width:
111                 if mouse_pos[1] > button.y and mouse_pos[1] < button.y+button.height:
112                     choice = i
113
114                     if choice >= 0:                      # Skill 선택
115                         message = "You choose " + player.skills[choice].name
116                     elif choice == -1:
117                         message = "Please select one"
118
119     # -----
120     # 화면 그리기
121     # -----
122     screen.fill((40,60,90))                  # 배경색
123
124     # 적 그리기
125     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
126     for i, enemy in enumerate(enemies):
127         x, y = left + i*gap, 120
128         pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
129         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
130         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
131
132     # 플레이어 그리기
133     pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
134     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
135     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
136
137     # 메시지 출력
```

Ln 107, Col 23 Spaces: 4 UTF-8 CRLF {} Python 3.13.7





```
C: > Users > sunje > Desktop > Commit > prac03.py > back_button

90 # -----
91 # 메인 메뉴 선택 (Fight / Run)
92 # -----
93 if level == 0:
94     for i, option in enumerate(options):
95         button = pygame.Rect(screen_width // 2 - 150, 150 + i * 50, 300, 50)
96         if mouse_pos[0] > button.x and mouse_pos[1] > button.y:
97             if mouse_pos[1] > button.y + 25:
98                 choice = i
99             if choice == 0:
100                 level, choice = 1, -1
101             elif choice == 1:
102                 message = "You can't run away"
103
104 # -----
105 # 스킬 선택
106 # -----
107 elif level == 1:
108     for i, skill in enumerate(player_skills):
109         button = pygame.Rect(screen_width // 2 - 150, 150 + i * 50, 300, 50)
110         if mouse_pos[0] > button.x and mouse_pos[1] > button.y:
111             if mouse_pos[1] > button.y + 25:
112                 choice = i
113             back_button = pygame.Rect(screen_width // 2 - 150, 250, 300, 50)
114             if mouse_pos[0] > back_button.x and mouse_pos[1] > back_button.y:
115                 if mouse_pos[1] > back_button.y + 25:
116                     choice = i+1
117
118             if choice == i+1:
119                 level, choice = 0, -1
120             elif choice >= 0:
121                 message = "You choose " + player_skills[choice]
122             elif choice == -1:
123                 message = "Please select one"
```

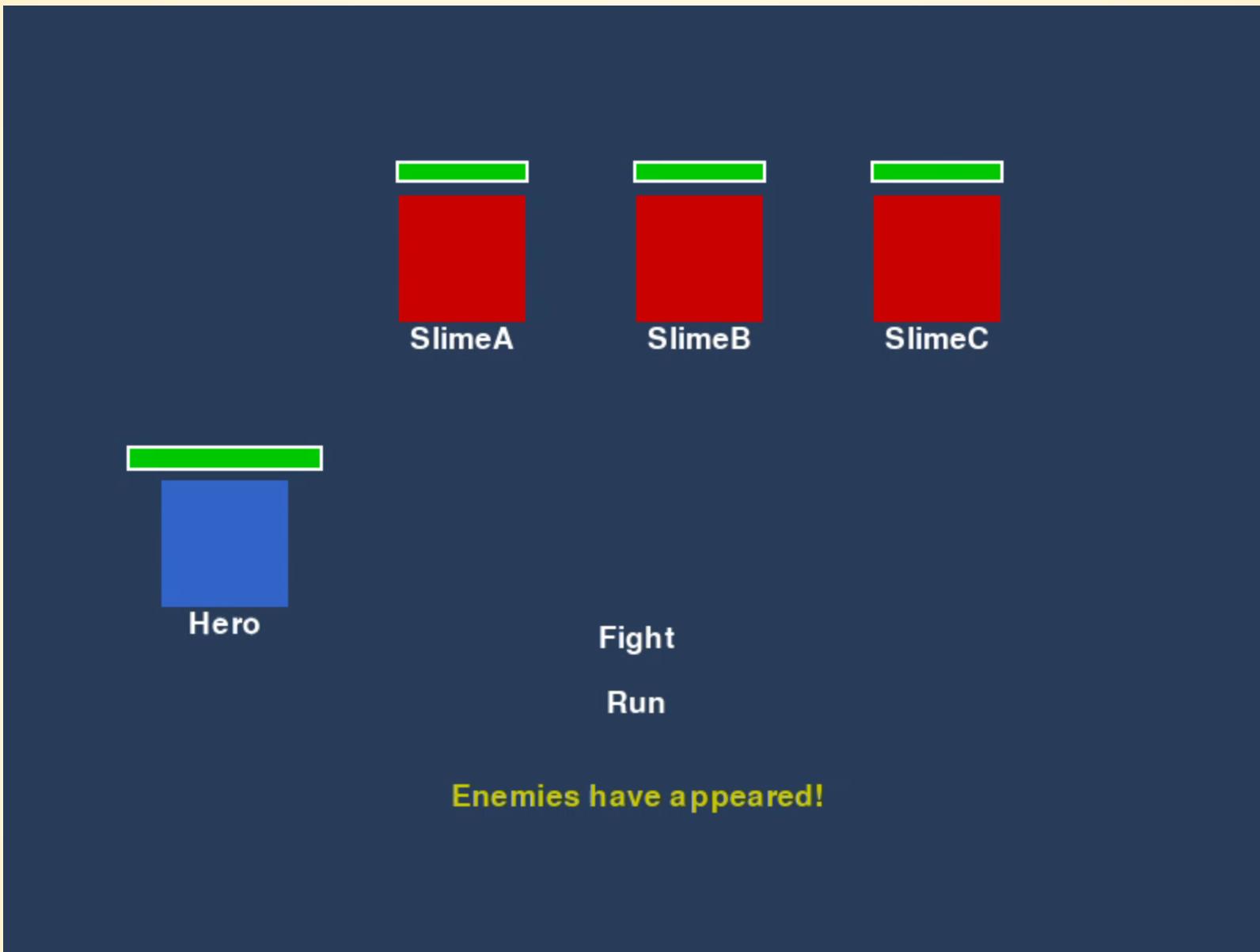
Ln 113, Col 28 Spaces: 4 UTF-8 CRLF {} Python 🐍 3.13.7 🔔

The screenshot shows a Python IDE interface with a dark theme. The top bar includes standard file operations like File, Edit, Selection, View, Go, Run, and a search bar. The left sidebar contains icons for file operations such as Open, Save, Find, and Preferences.

The main area displays a Python script named `prac03.py`. The code is color-coded for syntax highlighting. A red box highlights the line `draw_text(screen, font, "Back", (screen_width//2, 400+(i+1)*btn_gap), colors["WHITE"], True)`, which is part of a loop for drawing menu options. The script handles various game logic, including drawing enemies, the player, messages, and menu options. It uses Pygame functions like `fill`, `rect`, and `display.flip`.

```
File Edit Selection View Go Run ... ⏪ ⏩ Search
C: > Users > sunje > Desktop > Commit > prac03.py > ...
127     # -----
128     screen.fill((40,60,90))           # 배경색
129
130     # 적 그리기
131     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
132     for i, enemy in enumerate(enemies):
133         x, y = left + i*gap, 120
134         pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
135         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
136         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
137
138     # 플레이어 그리기
139     pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
140     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
141     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
142
143     # 메시지 출력
144     for i, line in enumerate(message.split("\n")):
145         draw_text(screen, font, line, (screen_width//2, 500+i*25), colors["YELLOW"], True)
146
147     # 메뉴 옵션 표시
148     if level == 0:
149         for i, option in enumerate(options):
150             draw_text(screen, font, option, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
151     elif level == 1:
152         for i, skill in enumerate(player.skills):
153             draw_text(screen, font, skill.name, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
154             draw_text(screen, font, "Back", (screen_width//2, 400+(i+1)*btn_gap), colors["WHITE"], True)
155
156     pygame.display.flip()           # 화면 업데이트
157     clock.tick(60)                # 초당 60프레임
158
159     pygame.quit()                 # 게임 종료
160
```

Bottom status bar: Ln 154, Col 38 Spaces: 4 UTF-8 CRLF {} Python 3.13.7



연습 문제 (3)

- 스킬을 선택 후 공격하기

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
1 # Created by 윤종현
2
3 import pygame, random
4
5 # -----
6 # 게임 초기화
7 # -----
8 pygame.init()                                     # pygame 라이브러리 초기화
9
10 screen_width, screen_height = 800, 600           # 화면 크기 설정
11 screen = pygame.display.set_mode((screen_width, screen_height))   # 화면 생성
12 pygame.display.set_caption("본인 학번, 이름")      # 게임 속도 조절용 시계
13 clock = pygame.time.Clock()                      # 글꼴 설정 (None = 기본 글꼴, 28 = 글자 크기)
14 font = pygame.font.Font(None, 28)
15
16 colors = {                                       # 자주 쓰는 색상 미리 정의
17     "WHITE": (255,255,255), "RED": (200,0,0), "BLUE": (50,100,200),
18     "GREEN": (0,200,0), "YELLOW": (200,200,0)
19 }
20
21 # -----
22 # 화면에 글자 그리는 함수
23 #
24 def draw_text(screen, font, text, pos, color, center=False):
25     # 글자를 화면에 렌더링
26     image = font.render(text, True, color)
27     # 글자 위치 지정
28     rectangle = image.get_rect(center=pos) if center else image.get_rect(topleft=pos)
29     # 화면에 붙이기
30     screen.blit(image, rectangle)
31
32 # -----
33 # 체력바 그리는 함수
34 # -----
```

C: > Users > sunje > Desktop > Commit > prac03.py > attack

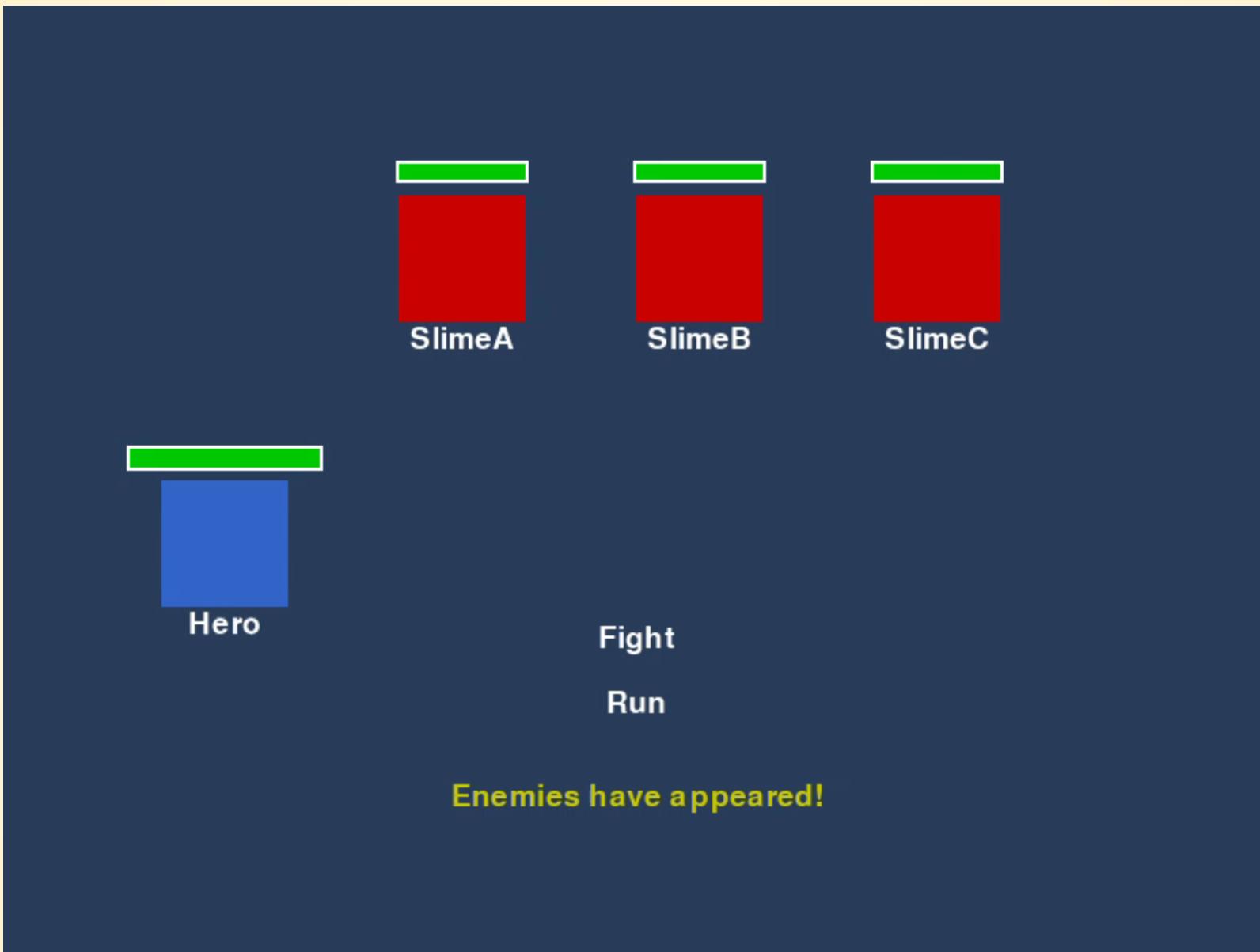
```
46     class Skill:
47         def __init__(self, name, power, hit):
48             self.name = name      # 기술 이름
49             self.power = power    # 공격력 배율
50             self.hit = hit        # 명중률 (%)
51
52     # -----
53 # 캐릭터/몬스터 클래스
54 # -----
55 class Creature:
56     def __init__(self, name, hp, attack, skills):
57         self.name = name      # 이름
58         self.max = hp          # 최대 체력
59         self.hp = hp            # 현재 체력
60         self.attack = attack    # 공격력
61         self.skills = skills    # 가지고 있는 기술 리스트
62
63     # 데미지를 입음
64     def take(self, damage):
65         self.hp = max(0, self.hp - damage) # 체력은 0 미만으로 내려가지 않음
66
67     # -----
68 # 공격 처리 함수
69 # -----
70 def attack(attacker, target, skill):
71     # 명중 체크
72     if random.randint(1, 100) > skill.hit:
73         return f"{attacker.name} missed!"
74     # 데미지 계산 (랜덤으로 80~120% 변동)
75     damage = int(attacker.attack * skill.power * random.uniform(0.8, 1.2))
76     # 대상 체력 감소
77     target.take(damage)
78     return f"{attacker.name} used {skill.name}! ({damage})"
79
```

prac03.py X

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
121      # -----#
122      # 스킬 선택
123      # -----
124      elif level == 1:
125          for i, skill in enumerate(player.skills):
126              button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
127              if mouse_pos[0] > button.x and mouse_pos[0] < button.x+button.width:
128                  if mouse_pos[1] > button.y and mouse_pos[1] < button.y+button.height:
129                      choice = i
130              back_button = pygame.Rect(screen_width//2-btn_width//2, 400+(i+1)*btn_gap-btn_height//2, btn_width, btn_height)
131              if mouse_pos[0] > back_button.x and mouse_pos[0] < back_button.x+back_button.width:
132                  if mouse_pos[1] > back_button.y and mouse_pos[1] < back_button.y+back_button.height:
133                      choice = i+1
134
135          if choice >= 0:                      # Skill 선택 후 Attack 버튼 선택 가능
136              attack_button = pygame.Rect(screen_width//2 + btn_width//2, 400+(i+1)*btn_gap-btn_height//2, btn_width, btn_height)
137              if mouse_pos[0] > attack_button.x and mouse_pos[0] < attack_button.x+attack_button.width:
138                  if mouse_pos[1] > attack_button.y and mouse_pos[1] < attack_button.y+attack_button.height:
139                      message = attack(player, enemies[0], player.skills[choice])    # 테스트로 첫번째 적 enemies[0]에게 공격
140
141          if choice == i+1:                   # Back 버튼 선택
142              level, choice = 0, -1
143
144          #elif choice >= 0:                 # Skill 선택
145          #    message = "You choose " + player.skills[choice].name
146          #    message = "Please select one"
147
148      # -----
149      # 화면 그리기
150      #
151      screen.fill((40,60,90))           # 배경색
152
153      # 적 그리기
154      gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
```

```
prac3.py X
C: > Users > sunje > Desktop > Commit > prac3.py > ...
152
153     # 적 그리기
154     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
155     for i, enemy in enumerate(enemies):
156         x, y = left + i*gap, 120
157         pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
158         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
159         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
160
161     # 플레이어 그리기
162     pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
163     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
164     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
165
166     # 메시지 출력
167     for i, line in enumerate(message.split("\n")):
168         draw_text(screen, font, line, (screen_width//2, 500+i*25), colors["YELLOW"], True)
169
170     # 메뉴 옵션 표시
171     if level == 0:
172         for i, option in enumerate(options):
173             draw_text(screen, font, option, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
174     elif level == 1:
175         for i, skill in enumerate(player.skills):
176             draw_text(screen, font, skill.name, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
177             draw_text(screen, font, "Back", (screen_width//2, 400+(i+1)*btn_gap), colors["WHITE"], True)
178             if choice >= 0:
179                 draw_text(screen, font, "Attack", (screen_width//2 + btn_width, 400+(i+1)*btn_gap), (200, 0, 200), True)
180
181     pygame.display.flip()                      # 화면 업데이트
182     clock.tick(60)                            # 초당 60프레임
183
184     pygame.quit()                            # 게임 종료
185
```



연습 문제 (4)

- 공격할 적 선택하기

2

8

四

1

8

1

prac03.py

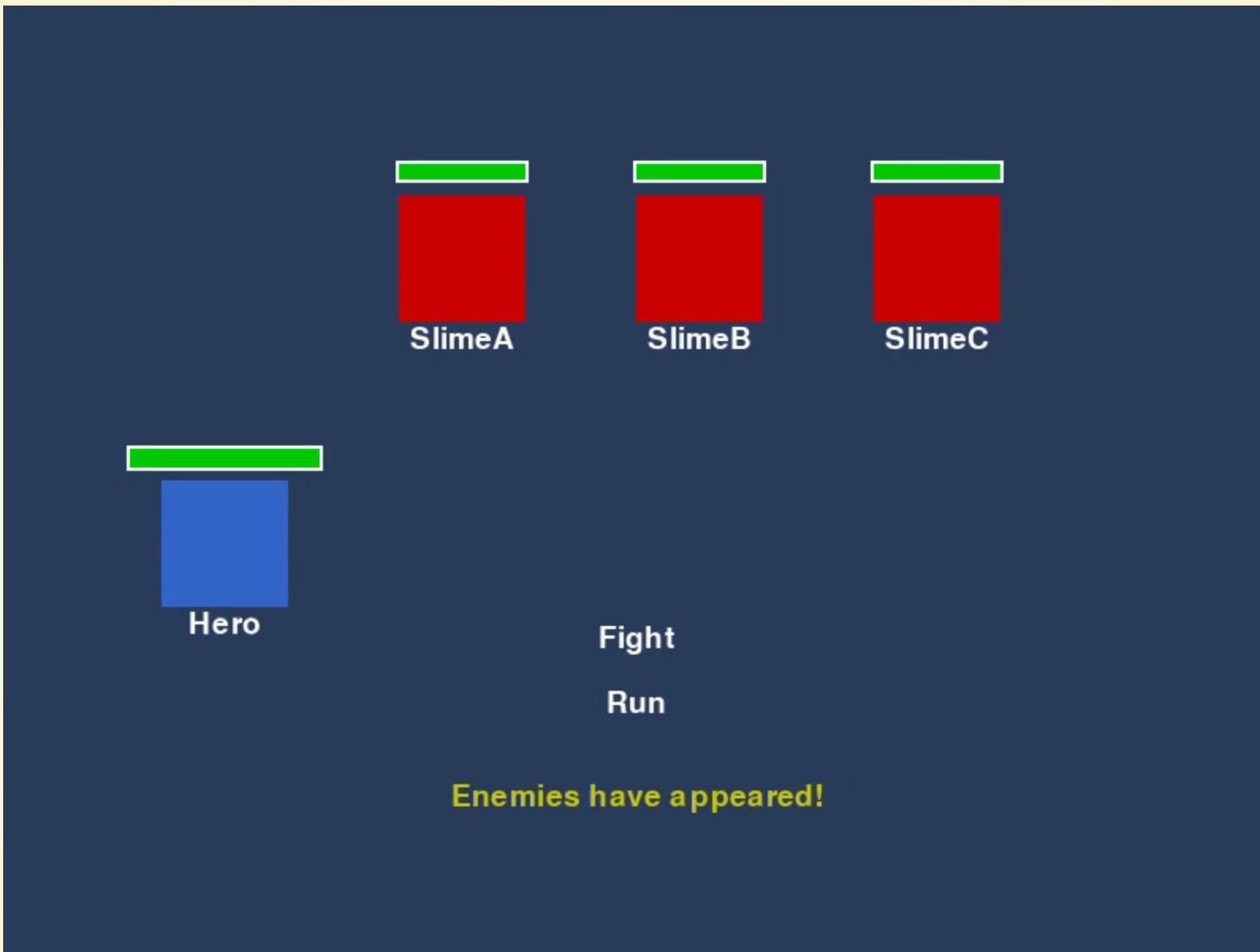
C: > Users > sunje > Desktop > Commit >  prac03.py >

```
121
122     # -----
123     # 스킬 선택
124     #
125     elif level == 1:
126         for i, skill in enumerate(player.skills):
127             button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
128             if mouse_pos[0] > button.x and mouse_pos[0] < button.x+button.width:
129                 if mouse_pos[1] > button.y and mouse_pos[1] < button.y+button.height:
130                     choice = i
131                     back_button = pygame.Rect(screen_width//2-btn_width//2, 400+(i+1)*btn_gap-btn_height//2, btn_width, btn_height)
132                     if mouse_pos[0] > back_button.x and mouse_pos[0] < back_button.x+back_button.width:
133                         if mouse_pos[1] > back_button.y and mouse_pos[1] < back_button.y+back_button.height:
134                             choice = i+1
135
136                     if choice >= 0:                      # Skill 선택 후 Attack 버튼 선택 가능
137                         attack_button = pygame.Rect(screen_width//2 + btn_width//2, 400+(i+1)*btn_gap-btn_height//2, btn_width, btn_height)
138                         if mouse_pos[0] > attack_button.x and mouse_pos[0] < attack_button.x+attack_button.width:
139                             if mouse_pos[1] > attack_button.y and mouse_pos[1] < attack_button.y+attack_button.height:
140                                 message = "You choose " + player.skills[choice].name + "\n Click one Slime!"
141                                 level = 2
142
143                     if choice == i+1:                  # Back 버튼 선택
144                         level, choice = 0, -1
145                     #elif choice >= 0:                  # Skill 선택
146                     #    message = "You choose " + player.skills[choice].name
147                     #elif choice == -1:
148                     #    message = "Please select one"
149
150     # -----
151     # 공격 대상 선택
152     #
153     elif level == 2:
154         gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
155         for i, enemy in enumerate(enemies):
```

prac03.py

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
149     # -----
150     # 공격 대상 선택
151     # -----
152     elif level == 2:
153         gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
154         for i, enemy in enumerate(enemies):
155             x, y = left + i*gap, 120
156             if mouse_pos[0] > x and mouse_pos[0] < x+80 and mouse_pos[1] > y and mouse_pos[1] < y+80:
157                 target = enemies[i]
158                 log = [attack(player, target, player.skills[choice])]    # 플레이어 공격
159                 message = "\n".join(log)
160                 level, choice = 0, -1
161
162     # -----
163     # 화면 그리기
164     # -----
165     screen.fill((40,60,90))                      # 배경색
166
167     # 적 그리기
168     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
169     for i, enemy in enumerate(enemies):
170         x, y = left + i*gap, 120
171         pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
172         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
173         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
174
175     # 플레이어 그리기
176     pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
177     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
178     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
179
180     # 메시지 출력
181     for i, line in enumerate(message.split("\n")):
182         draw_text(screen, font, line, (screen_width//2, 500+i*25), colors["YELLOW"], True)
```



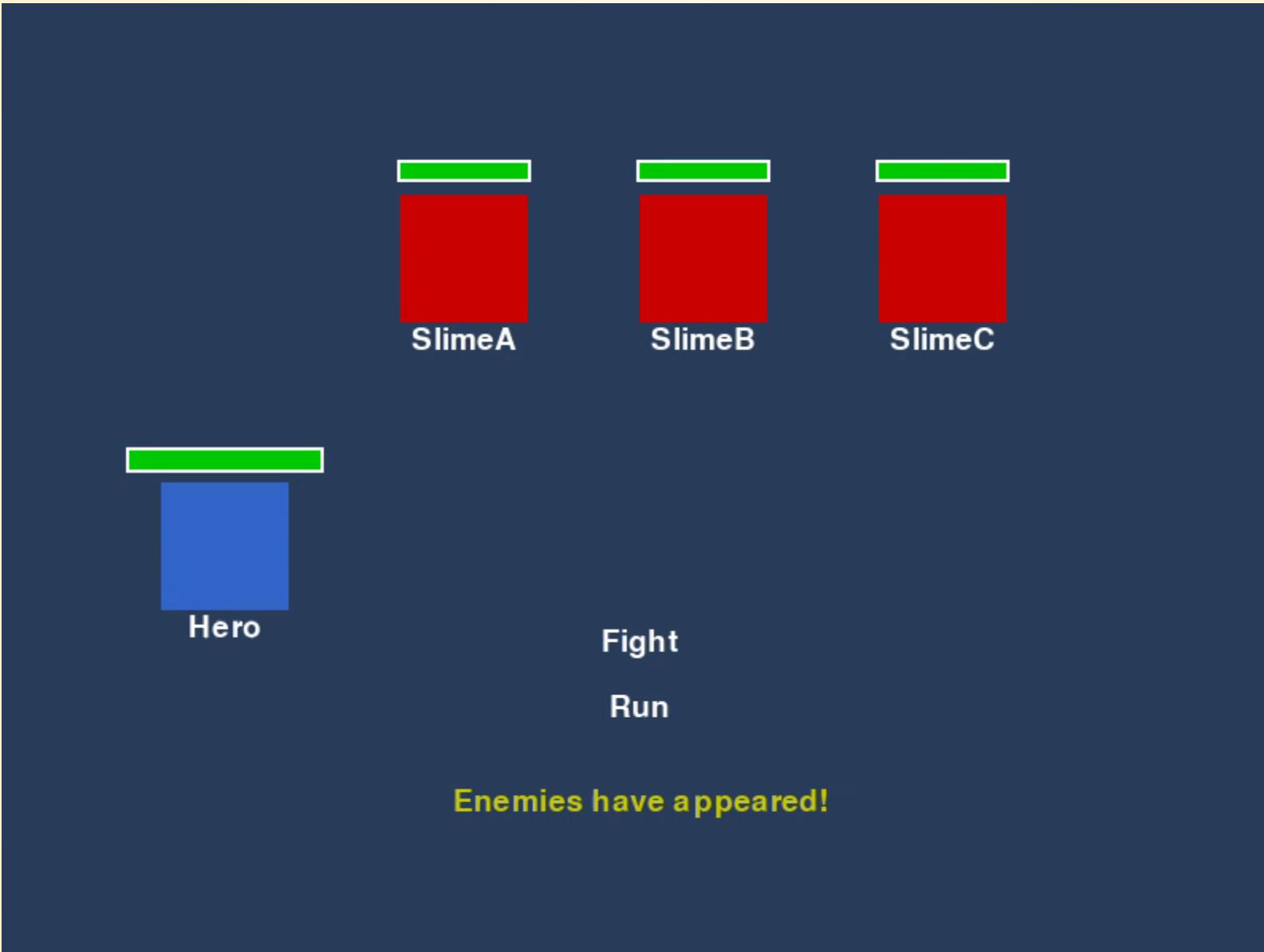
연습 문제 (5)

- 적도 플레이어를 공격하기

prac03.py x

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
149     # -----
150     # 공격 대상 선택
151     # -----
152     elif level == 2:
153         gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
154         for i, enemy in enumerate(enemies):
155             x, y = left + i*gap, 120
156             if mouse_pos[0] > x and mouse_pos[0] < x+80 and mouse_pos[1] > y and mouse_pos[1] < y+80:
157                 target = enemies[i]
158                 log = [attack(player, target, player.skills[choice])]          # 플레이어가 적을 공격
159                 for enemy in enemies:
160                     log.append(attack(enemy, player, random.choice(enemy.skills))) # 적이 플레이어를 공격
161                 message = "\n".join(log)
162                 level, choice = 0, -1
163
164     # -----
165     # 화면 그리기
166     # -----
167     screen.fill((40,60,90))           # 배경색
168
169     # 적 그리기
170     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
171     for i, enemy in enumerate(enemies):
172         x, y = left + i*gap, 120
173         pygame.draw.rect(screen, colors["RED"], (x, y, 80, 80))
174         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
175         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
176
177     # 플레이어 그리기
178     pygame.draw.rect(screen, colors["BLUE"], (100,300,80,80))
179     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
180     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
181
182     # 메시지 출력
```



연습 문제 (6)

- 플레이어와 적이 살아있는지 판별 후 승리/패배 결정하기

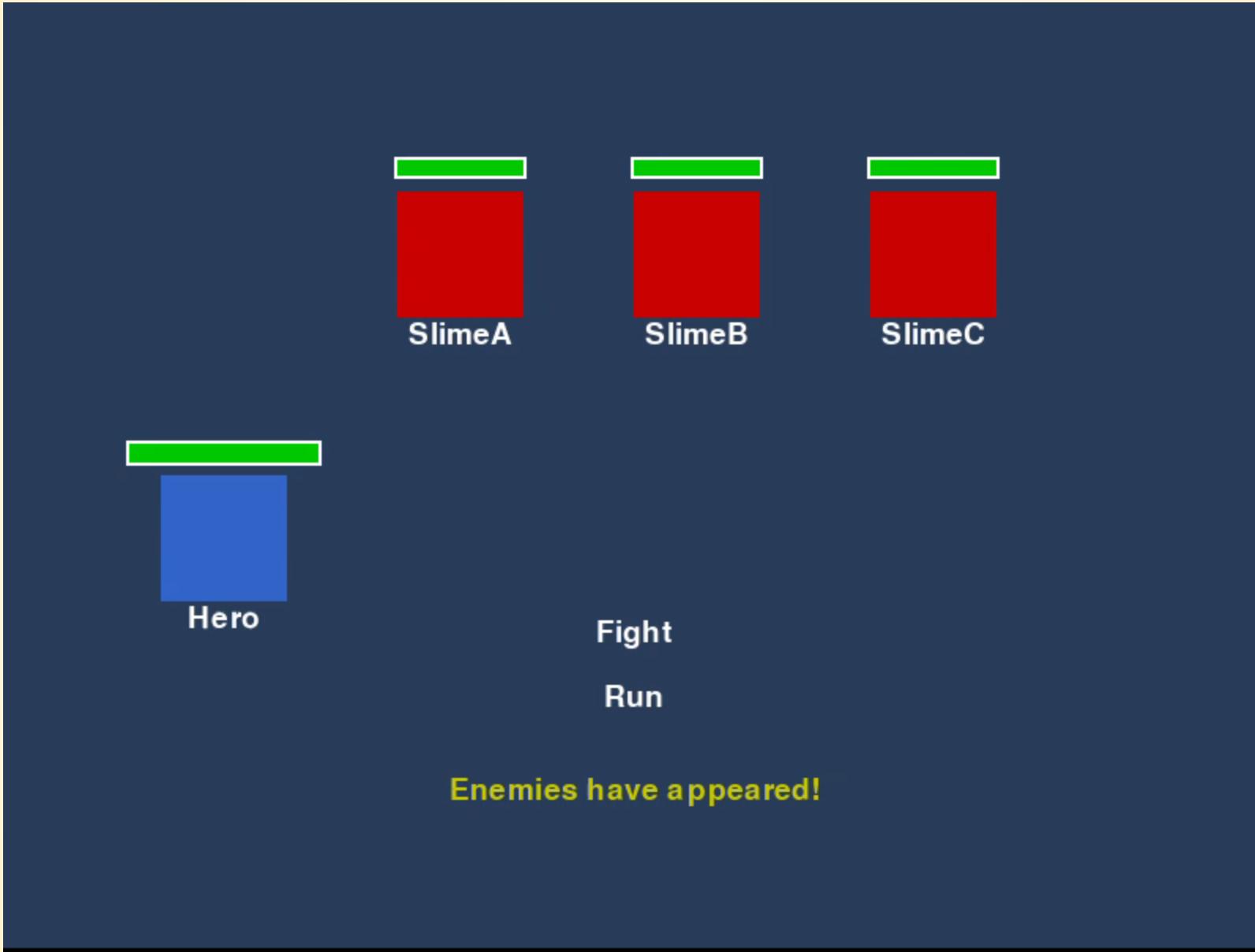
C: > Users > sunje > Desktop > Commit > prac03.py > Creature > alive

```
52     # -----
53     # 캐리터/몬스터 클래스
54     # -----
55     class Creature:
56         def __init__(self, name, hp, attack, skills):
57             self.name = name      # 이름
58             self.max = hp        # 최대 체력
59             self.hp = hp         # 현재 체력
60             self.attack = attack # 공격력
61             self.skills = skills # 가지고 있는 기술 리스트
62
63         # 데미지를 입음
64         def take(self, damage):
65             self.hp = max(0, self.hp - damage) # 체력은 0 미만으로 내려가지 않음
66
67         # 살아있는지 확인
68         def alive(self):
69             return self.hp > 0
70
71         # -----
72         # 공격 처리 함수
73         # -----
74         def attack(attacker, target, skill):
75             # 명중 체크
76             if random.randint(1, 100) > skill.hit:
77                 return f"{attacker.name} missed!"
78             # 데미지 계산 (랜덤으로 80~120% 변동)
79             damage = int(attacker.attack * skill.power * random.uniform(0.8, 1.2))
80             # 대상 체력 감소
81             target.take(damage)
82             return f"{attacker.name} used {skill.name}! ({damage})"
83
84         # -----
85         # 게임 실행
```

prac03.py

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
168     # -----
169     # 화면 그리기
170     #
171     screen.fill((40,60,90))           # 배경색
172
173     # 적 그리기
174     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
175     for i, enemy in enumerate(enemies):
176         x, y = left + i*gap, 120
177         pygame.draw.rect(screen, colors["RED"] if enemy.alive() else colors["WHITE"], (x, y, 80, 80))
178         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
179         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
180
181     # 플레이어 그리기
182     pygame.draw.rect(screen, colors["BLUE"] if player.alive() else (100,100,100), (100,300,80,80))
183     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
184     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
185
186     # 메시지 출력
187     for i, line in enumerate(message.split("\n")):
188         draw_text(screen, font, line, (screen_width//2, 500+i*25), colors["YELLOW"], True)
189
190     # 메뉴 옵션 표시
191     if level == 0:
192         for i, option in enumerate(options):
193             draw_text(screen, font, option, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
194     elif level == 1:
195         for i, skill in enumerate(player.skills):
196             draw_text(screen, font, skill.name, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
197             draw_text(screen, font, "Back", (screen_width//2, 400+(i+1)*btn_gap), colors["WHITE"], True)
198             if choice >= 0:
199                 draw_text(screen, font, "Attack", (screen_width//2 + btn_width, 400+(i+1)*btn_gap), (200, 0, 200), True)
200
201     pygame.display.flip()           # 화면 업데이트
```



prac03.py X

C: > Users > sunje > Desktop > Commit > prac03.py > [o] gameover

```
97     # -----
98     # 게임 루프
99     #
100    options = ["Fight", "Run"]           # 메뉴 버튼 글자
101    choice, level = -1, 0              # 선택한 메뉴 (choice=0 첫번째 메뉴, choice=2 두번째 메뉴, ...), 메뉴 레벨
102    message = "Enemies have appeared!" # 메시지 출력
103    btn_width, btn_height, btn_gap = 60, 30, 40      # 메뉴 버튼 크기, 간격 (변경 가능)
104    running, gameover = True, False
105
106    while running:
107        for event in pygame.event.get():
108            if event.type == pygame.QUIT:          # 종료 버튼 누르면 게임 종료
109            running = False
110            if event.type == pygame.MOUSEBUTTONDOWN and not gameover:    # 마우스 버튼이 눌렸을 때
111                mouse_pos = pygame.mouse.get_pos()      # 마우스의 포지션을 받아서 mouse_pos에 저장
112
113                # 메인 메뉴 선택 (Fight / Run)
114
115                if level == 0:
116                    for i, option in enumerate(options):
117                        button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
118                        if mouse_pos[0] > button.x and mouse_pos[0] < button.x+button.width:
119                            if mouse_pos[1] > button.y and mouse_pos[1] < button.y+button.height:
120                                choice = i
121                                if choice == 0:                      # Fight 메뉴 선택
122                                    level, choice = 1, -1
123                                elif choice == 1:                  # Run 메뉴 선택
124                                    message = "You can't run away!"
125
126                # 스킬 선택
127
128                elif level == 1:
129                    for i, skill in enumerate(player.skills):
130                        button = pygame.Rect(screen_width//2-btn_width//2, 400+i*btn_gap-btn_height//2, btn_width, btn_height)
```



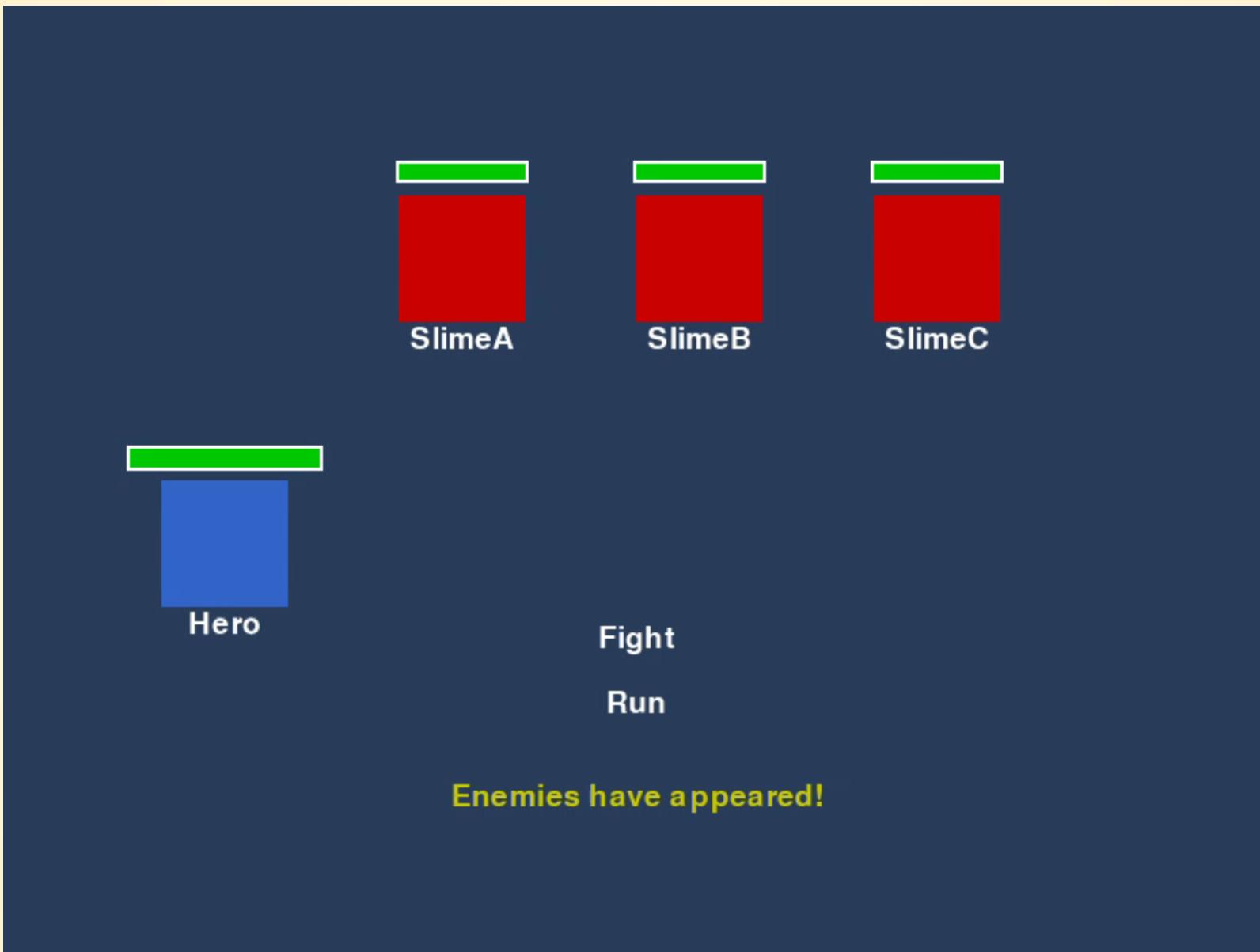
prac03.py X

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
153     # -
154     # 공격 대상 선택
155     #
156     elif level == 2:
157         gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
158         for i, enemy in enumerate(enemies):
159             x, y = left + i*gap, 120
160             if mouse_pos[0] > x and mouse_pos[0] < x+80 and mouse_pos[1] > y and mouse_pos[1] < y+80:
161                 target = enemies[i]
162                 log = [attack(player, target, player.skills[choice])]          # 플레이어가 적을 공격
163                 for enemy in enemies:
164                     log.append(attack(enemy, player, random.choice(enemy.skills))) # 적이 플레이어를 공격
165                 message = "\n".join(log)
166                 level, choice = 0, -1
167                 # 승리/패배 체크
168                 if not player.alive():
169                     message += " ==> Defeat...T_T"
170                     gameover = True
171                 elif not any(enemy.alive() for enemy in enemies):
172                     message += " ==> Victory!!!"
173                     gameover = True
174
175     # -
176     # 화면 그리기
177     #
178     screen.fill((40,60,90))           # 배경색
179
180     # 적 그리기
181     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
182     for i, enemy in enumerate(enemies):
183         x, y = left + i*gap, 120
184         pygame.draw.rect(screen, colors["RED"] if enemy.alive() else colors["WHITE"], (x, y, 80, 80))
185         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
186         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
```

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
180     # 적 그리기
181     gap, left = 150, (screen_width - (len(enemies)-1) * 150) // 2
182     for i, enemy in enumerate(enemies):
183         x, y = left + i*gap, 120
184         pygame.draw.rect(screen, colors["RED"] if enemy.alive() else colors["WHITE"], (x, y, 80, 80))
185         draw_text(screen, font, enemy.name, (x+40, y+90), colors["WHITE"], True)
186         draw_hp(screen, x, y-20, 80, 10, enemy.hp, enemy.max, colors)
187
188     # 플레이어 그리기
189     pygame.draw.rect(screen, colors["BLUE"] if player.alive() else (100,100,100), (100,300,80,80))
190     draw_text(screen, font, player.name, (140,390), colors["WHITE"], True)
191     draw_hp(screen, 80, 280, 120, 12, player.hp, player.max, colors)
192
193     # 메시지 출력
194     for i, line in enumerate(message.split("\n")):
195         draw_text(screen, font, line, (screen_width//2, 500+i*25), colors["YELLOW"], True)
196
197     # 메뉴 옵션 표시
198     if not gameover:
199         if level == 0:
200             for i, option in enumerate(options):
201                 draw_text(screen, font, option, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
202         elif level == 1:
203             for i, skill in enumerate(player.skills):
204                 draw_text(screen, font, skill.name, (screen_width//2, 400+i*btn_gap), (0,200,200) if i==choice else colors["WHITE"], True)
205             draw_text(screen, font, "Back", (screen_width//2, 400+(i+1)*btn_gap), colors["WHITE"], True)
206             if choice >= 0:
207                 draw_text(screen, font, "Attack", (screen_width//2 + btn_width, 400+(i+1)*btn_gap), (200, 0, 200), True)
208
209             pygame.display.flip()                      # 화면 업데이트
210             clock.tick(60)                           # 초당 60프레임
211
212     pygame.quit()                            # 게임 종료
```



연습 문제 (7)

- 승률을 높이도록 밸런싱을 해보시오.
 - 예) 적들의 명중률(hit)을 낮추기

C: > Users > sunje > Desktop > Commit > prac03.py > ...

```
71     # -----
72     # 공격 처리 함수
73     #
74     def attack(attacker, target, skill):
75         # 명중 체크
76         if random.randint(1, 100) > skill.hit:
77             return f"{attacker.name} missed!"
78         # 데미지 계산 (랜덤으로 80~120% 변동)
79         damage = int(attacker.attack * skill.power * random.uniform(0.8, 1.2))
80         # 대상 체력 감소
81         target.take(damage)
82         return f"{attacker.name} used {skill.name}! ({damage})"
83
84     # -----
85     # 게임 실행
86     #
87     # 플레이어 캐릭터 생성
88     player = Creature("Hero", 50, 5, [Skill("Slash", 4, 85), Skill("Fire", 6, 75)])
89
90     # 적 캐릭터 생성
91     enemies = [
92         Creature("SlimeA", 30, 3, [Skill("Bite", 3, 40)]),
93         Creature("SlimeB", 25, 3, [Skill("Bite", 3, 45)]),
94         Creature("SlimeC", 20, 2, [Skill("Bite", 2, 50)])]
95     ]
96
97     # -----
98     # 게임 루프
99     #
100    options = ["Fight", "Run"]                                # 메뉴 버튼 글자
101    choice, level = -1, 0                                     # 선택한 메뉴 (choice=0 첫번째 메뉴, choice=2 두번째 메뉴, ...), 메뉴 레벨
102    message = "Enemies have appeared!"                      # 메시지 출력
103    btn_width, btn_height, btn_gap = 60, 30, 40              # 메뉴 버튼 크기, 간격 (변경 가능)
104    running, gameover = True, False
```

