



# Busan science high school

## 2023 Ocean ICT Festival

## 2023 BOIF

B  
21

QR 코드 영역  
QR 삽입 후  
테두리 삭제

Youtube 영상 QR

## <해양 환경오염과 화학분석>

Great & Gorgeous Girls  
2202 이윤이 2204 정지안

### 작품 선정 동기

- 선정 동기
  - 심각한 해양 환경오염
  - 해양 생태계와 인류의 건강에 부정적인 영향을 미칠 수 있는 다양한 오염물질의 발견
- 해결 방법
  - ① 해양 오염 물질의 종류와 양 파악
  - ② 오염물질의 종류에 따른 농도 기준치를 설정 및 코드를 통한 오염수의 화학적 분석
  - ③ 분석 결과에 따라 기준 초과 시 오염수의 유입 차단

### 이론적 배경

- 원자 흡수분광법 (Atomic Absorption Spectrometry, AAS)
  - 금속 또는 금속 이온 등 해양 환경오염물질의 양적 분석에 활용
  - 수은(Hg), 납(Pb), 카드뮴(Cd), 구리(Cu) 등의 정량 측정 및 모니터링 가능
- 플루오린세스 분광법 (Fluorescence Spectrometry)
  - 유기오염물질 분석
  - 흡수, 활성화의 단계에서 흡수 스펙트럼 기록 후 흡수된 에너지로 오염물질 내 전자 활성화
  - 발광, 검출의 단계에서 전자의 발광 스펙트럼을 플루오린세스 분광계가 감지해 정량 측정

### 해양환경오염 방지를 위한 물 유입 여부 코드

\* 본 작품은 pygame, sys 모듈을 활용하여 코드를 작성하였다. 작품의 원활한 설명을 위해 주요 단계의 코드에 대해서만 다루겠다.

#### ■ Step 1. 초기 설정하기

```
import pygame
import sys

# Pygame 초기화
pygame.init()

# 화면 크기 설정
screen_width = 1152
screen_height = 597
screen = pygame.display.set_mode((screen_width, screen_height))
pygame.display.set_caption("World Ocean Pollution Simulating Program")

# 배경 이미지 로드
background = pygame.image.load("D:/2학년/수학/과제인공지능/worldocean.png") # 이미지 파일 경로를 적절하게 수정하세요
background_rect = background.get_rect()

# 확대 비율
zoom_factor = 2

# 바다 이름 위치 및 정보 (북극해, 남극, 대서양, 인도양, 태평양)
ocean_locations = {
    "북극해": (921, 21),
    "대서양": (330, 211),
    "인도양": (1074, 185),
    "인도양": (745, 385),
    "남극": (599, 518)
}
```

```
# 버튼 생성 함수
def create_button(rect, text):
    font = pygame.font.Font(None, 36)
    button = pygame.Surface((rect.width, rect.height))
    button.fill((255, 255, 255))
    text_surface = font.render(text, True, (0, 0, 0))
    text_rect = text_surface.get_rect(center=(rect.width // 2, rect.height // 2))
    button.blit(text_surface, text_rect)
    return button

buttons = []
for ocean, (x, y) in ocean_locations.items():
    button_rect = pygame.Rect(x, y, 100, 40)
    buttons[ocean] = {
        "rect": button_rect,
        "button": create_button(button_rect, ocean)
    }

# 중금속과 유기물질 양
metal_amount = 0
organic_amount = 0

# 초기 바다 설정
current_ocean = None

# Pygame 대화창 초기화
input_dialog = None
input_font = pygame.font.Font(None, 36)
input_text = ""

# 결과가 표시된 시간을 추적하는 변수
result_show_time = None
```

#### ■ Step 2. 게임 루프 설계하기

- 특정 바다를 선택하면 입력창이 뜨고 오염물질의 값을 입력
- 마우스를 한 번 더 클릭하면 기준치와 입력값을 비교하여 물 유입 여부를 결정
- 확대되었을 때 지역 정보 초기화 및 선택된 지역의 이름을 저장하도록 코드 설계

```
while running:
    result_show_time = None # 결과가 표시되지 않은 상태로 초기화

    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False
        elif event.type == pygame.MOUSEBUTTONDOWN:
            if event.button == 1:
                for ocean, data in buttons.items():
                    if event.data in data["rect"].collidoint(event.pos):
                        if zoomed:
                            screen.blit(background, (0, 0))
                            zoomed = False
                            current_ocean = None # 확대되었을 때 지역 정보 초기화
                            if input_dialog.open:
                                input_dialog = None # 입력창 닫기
                                input_dialog.open = False
                        else:
                            # 확대되는 이미지의 크기와 중심을 버튼 위치로 조정
                            zoomed_image = pygame.transform.scale(background, (screen_width * zoom_factor, screen_height * zoom_factor))
                            zoomed_image_rect = zoomed_image.get_rect(center=data["rect"].center)

                            screen.blit(zoomed_image, (0, 0))
                            if not input_dialog.open:
                                input_dialog = pygame.Rect(200, 200, 400, 200)
                                input_text = ""
                                current_ocean = ocean # 선택된 지역의 이름을 저장
                                input_dialog.open = True
                            pygame.display.flip()
```

```
elif event.type == pygame.KEYDOWN:
    if input_dialog is not None:
        if event.key == pygame.K_RETURN:
            # 입력 받은 값을 변수에 저장
            input_values = input_text.split()
            if len(input_values) != 2:
                metal_amount, organic_amount = map(int, input_values)
            if result_show_time is None: # 결과가 아직 표시되지 않은 경우에만
                print(current_ocean, "물을 유입합니다.")
                print("중금속 양:", metal_amount, "ng/kg")
                print("유기오염물질 양:", organic_amount, "ng/g")
                if metal_amount <= 100 and organic_amount <= 50:
                    print(current_ocean, "안전합니다. 계속 흘러갑니다.")
                else:
                    print(current_ocean, "위험! 검문이 필요합니다.")
                    result_show_time = pygame.time.get_ticks() # 결과를 보여줄 시간을 지정
            input_dialog = None
            input_dialog.open = False
            input_text = "" # 입력 초기화
        elif event.key == pygame.K_BACKSPACE:
            input_text = input_text[:-1]
        else:
            input_text += event.unicode
```

#### ■ Step 3. 실행하기

- 'Run'을 클릭하면 아래 사진의 팝업창이 나타남
- 다섯 개의 바다 중 오염수를 유입시키고자 하는 곳의 버튼을 누르면 입력창이 나타남
- 입력창에 오염물질(중금속, 유기오염물질)의 양을 입력
- Jupyter Notebook 상에서 오염수의 유입 결정



(팝업창)

```
screen.fill((0, 0, 0)) # 배경을 검은색으로 채우기

if zoomed:
    # 중심이 버튼 위치로 조정된 이미지를 화면에 표시
    screen.blit(zoomed_image, (screen_width // 2 - zoomed_image_rect.centerx, screen_height // 2 - zoomed_image_rect.centery))
else:
    screen.blit(background, (0, 0)) # 초기 이미지 표시

for data in buttons.values():
    screen.blit(data["button"], data["rect"].toleft())

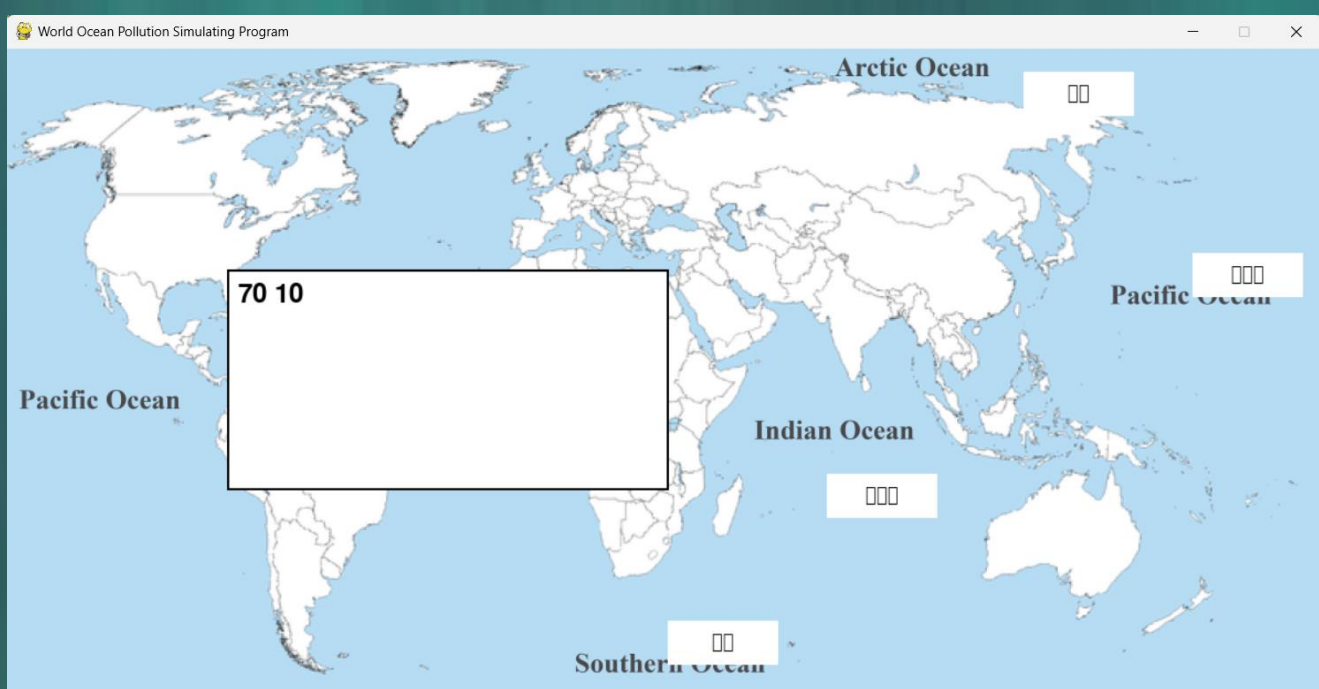
if input_dialog.open:
    pygame.draw.rect(screen, (255, 255, 255), input_dialog)
    pygame.draw.rect(screen, (0, 0, 0), input_dialog, 2)
    input_surface = input_font.render(input_text, True, (0, 0, 0))
    screen.blit(input_surface, (input_dialog.x + 10, input_dialog.y + 10))

if current_ocean and not input_dialog.open and result_show_time is not None:
    if pygame.time.get_ticks() - result_show_time >= 3000:
        result_show_time = None

pygame.display.update()
clock.tick(60)

pygame.quit()
sys.exit()
```

EX 1. 대서양에 중금속 70mg/kg, 유기오염물질 10ng/g을 입력했을 때



대서양 물을 유입합니다.  
중금속 양: 70 mg/kg  
유기오염물질 양: 10 ng/g  
대서양 안전합니다. 계속 흘러갑니다.

EX 2. 태평양에 중금속 150mg/kg, 유기오염물질 40ng/g을 입력했을 때



태평양 물을 유입합니다.  
중금속 양: 150 mg/kg  
유기오염물질 양: 40 ng/g  
태평양 위험! 검문이 필요합니다.

### 기대효과

- ① 화학 오염물질의 종류와 양에 따라 유입 여부 판단
- ② 해양 환경오염의 심화 방지
- ③ 해양 환경오염 대응 및 예방
- ④ 해양 환경오염의 심각성에 대한 경고