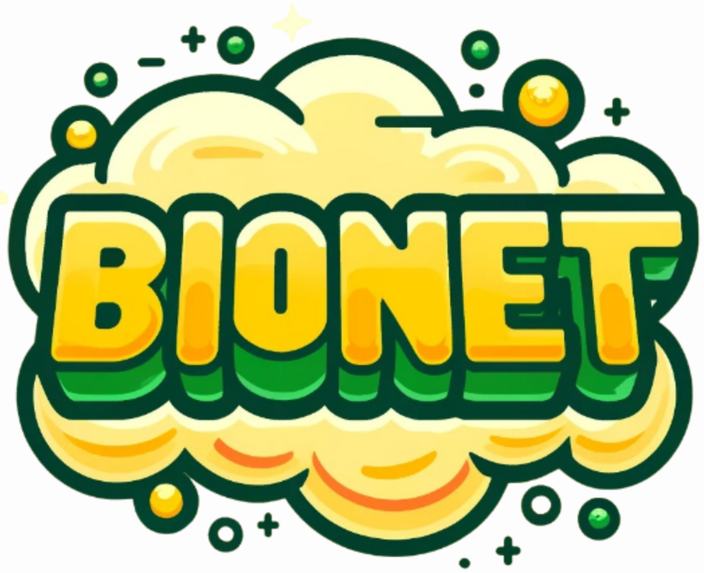
khack 2024

생태계 시뮬레이션 게임
BioNet

2024.5.18 Khack

김오이임
김채은 오윤상 이정희 임창현



이란?

LLM을 이용한 생태계 시뮬레이션 게임

개체 간 상호 작용을
LLM이 처리

BioNet이란?

사용자의 개입 → 생물 다양성을 구현 or 관찰
생태계 보존에 흥미 유발

BioNet이란?

무슨 상호 작용?

ChatGPT

교배

포식/피식

BioNet이란?

무슨 상호 작용?

교배

LLM이 동물 A, B의 주요한
특성을 반영한 자식 생성

ex 호랑이 + 펭귄 → 바다표범

BioNet이란?

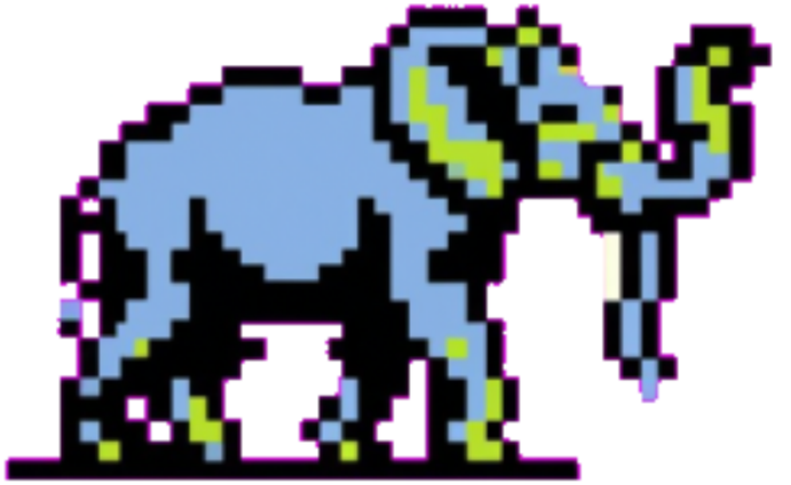
교배 구현

```python
#두 동물 사이 한 동물
a = "Orca"
b = "dolphin"
region = "ocean"
stream = client.chat.completions.create(
  model="solar-1-mini-chat",
  messages=[
    {
      "role": "system",
      "content": "You must answer as only one word form. you must not reuse any word in my question"
    },
    {
      "role": "user",
      "content": f"what is the another animal in {region} between {a} and {b}"
    }
  ],
  stream=True,
)

for chunk in stream:
  if chunk.choices[0].delta.content is not None:
    print(chunk.choices[0].delta.content, end="")
```

```
whale
```

재미있는 점: 환경 반영!

```python
# 텍스트 프롬프트를 사용하여 이미지 생성
response = openai.Image.create(
  prompt=f"{animal} with 8 bit image like old-pokemon style graphic, background color is complement color",
  n=1,
  size="256x256"
)

# 생성된 이미지 URL 가져오기blob:https://colab.research.google.com/9e1fc036-bb7d-4a9b-9a23-fe8634ebafbc
image_url = response['data'][0]['url']
print("Generated image URL:", image_url)

image_response = requests.get(image_url)

# 이미지 파일로 저장
with open("generated_image.png", "wb") as f:
    f.write(image_response.content)

print("Image downloaded and saved as 'generated_image.png'")


input = Image.open('generated_image.png') # load image

# 입력 이미지 경로 및 출력 이미지 경로
input_path = 'generated_image.png'
output_path = 'generated_image_1.png'

image = Image.open(input_path).convert("RGBA")

# 제거할 배경 색상 (여기서는 흰색으로 가정, 필요에 따라 변경)
datas = image.getdata()
background_color = list(datas[0])[:3]

# 배경 색상을 투명하게 변경
new_data = []
for item in datas:
    color = list(item[:3])
    if ((item[0]-background_color[0])**2+(item[1]-background_color[1])**2+(item[2]-background_color[2])**2)**0.5<50:
        new_data.append((255, 255, 255, 0))  # 완전 투명
    else:
        new_data.append(item)

# 새로운 이미지 데이터를 적용
image.putdata(new_data)
```



BioNet이란?

무슨 상호 작용?

동물 A, B 중
누가 먹이사슬의 위에
위치하는지 LLM이 판정

포식/피식

BioNet이란?

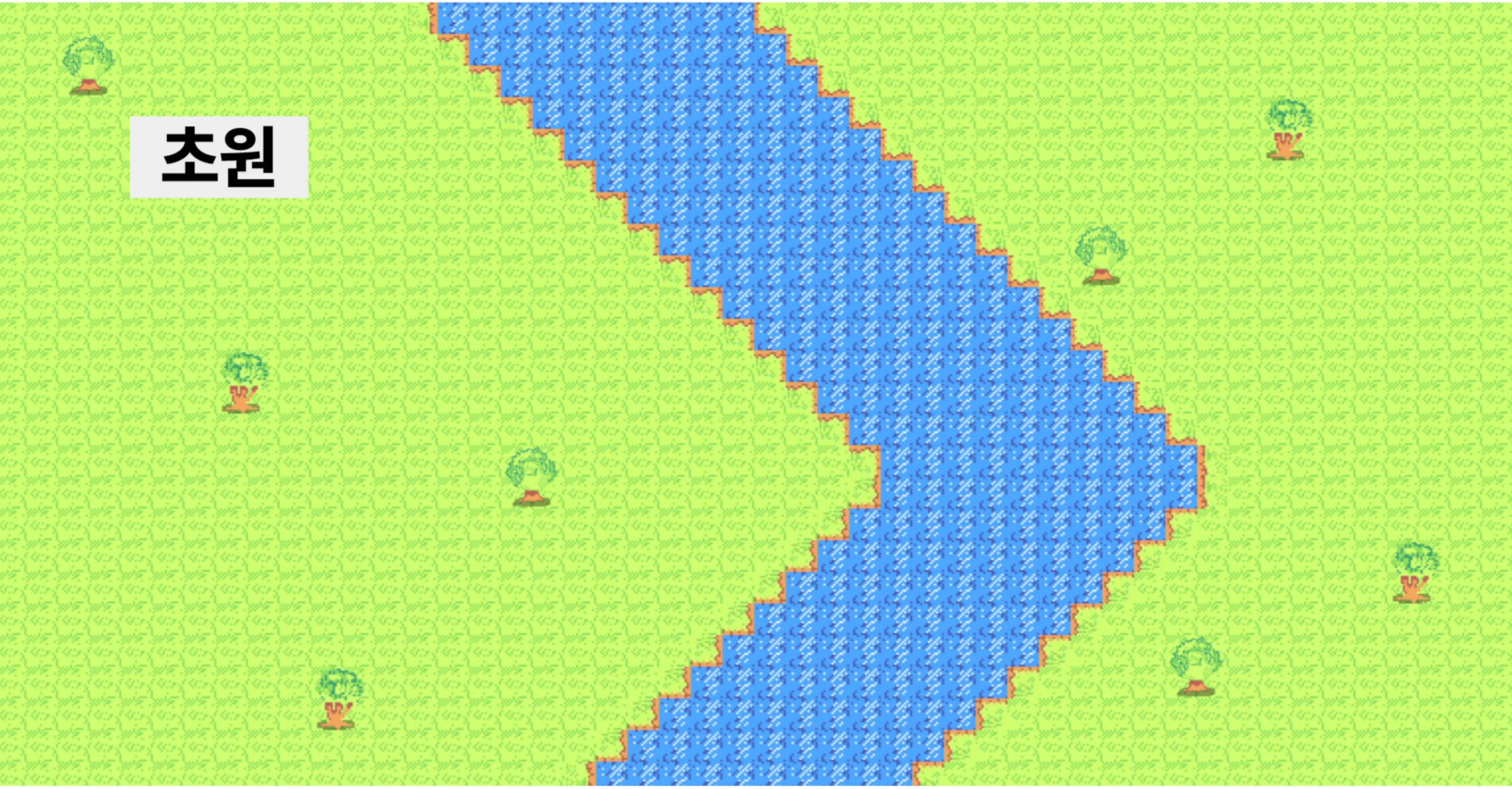
포식/피식 관계 구현

```
client = OpenAI(
  api_key="up_k2nsg5SPYONKzC8V0th2ldEblE0vz",
  base_url="https://api.upstage.ai/v1/solar"
)

#a가 b를 사냥할수 있는지
def can_hunt(a,b):
  stream = client.chat.completions.create(
    model="solar-1-mini-chat",
    messages=[
      {
        "role": "system",
        "content": "You must answer as only 'True' or 'False'"
      },
      {
        "role": "user",
        "content": f"Can {a} hunt {b}?"
      }
    ],
    stream=True,
  )

  for chunk in stream:
    if chunk.choices[0].delta.content is not None:
      return chunk.choices[0].delta.content

print(can_hunt('tiger', 'dear'))
```

```
True
```

게임 요소

맵


초원

사막

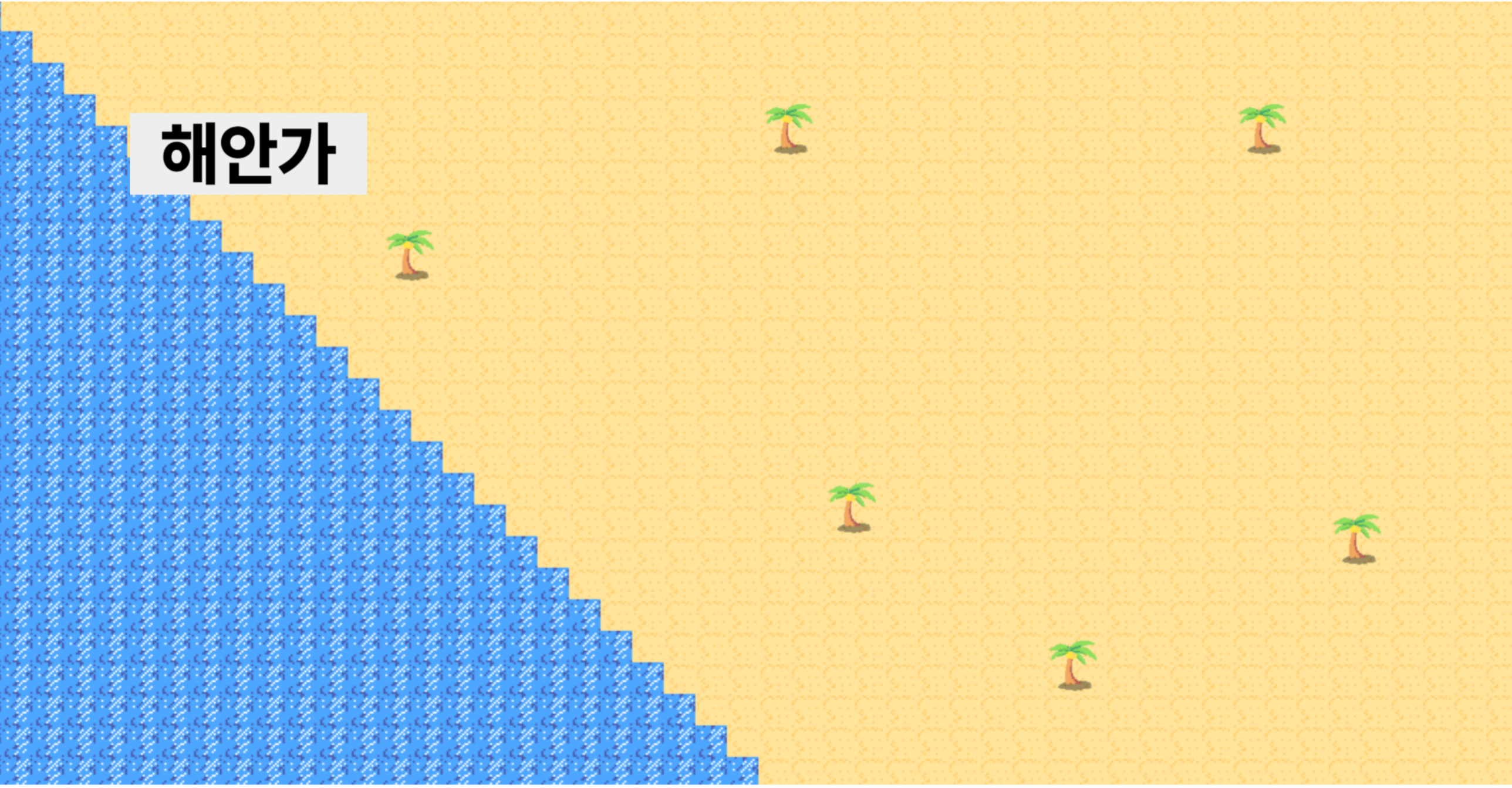


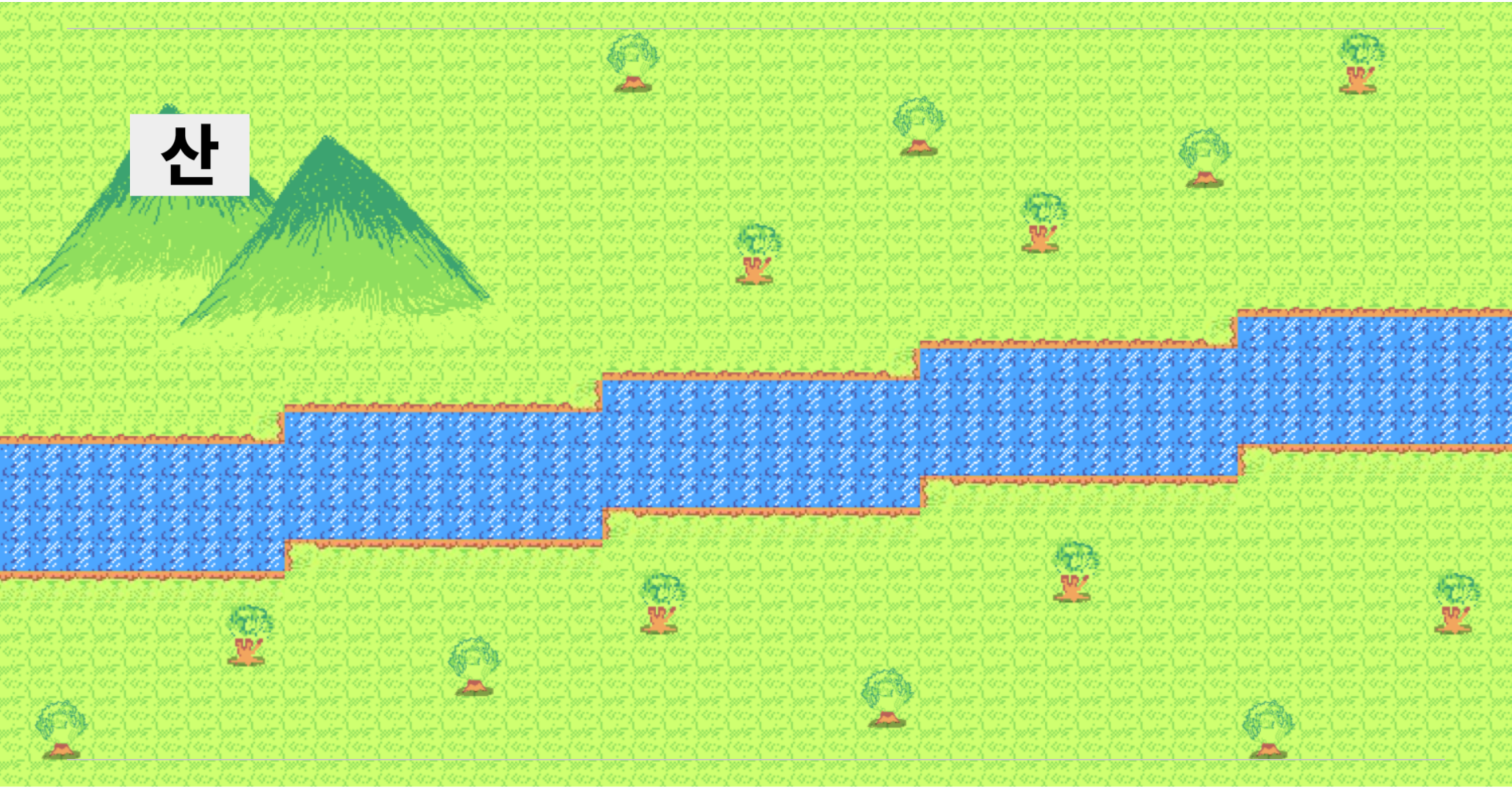






해안가



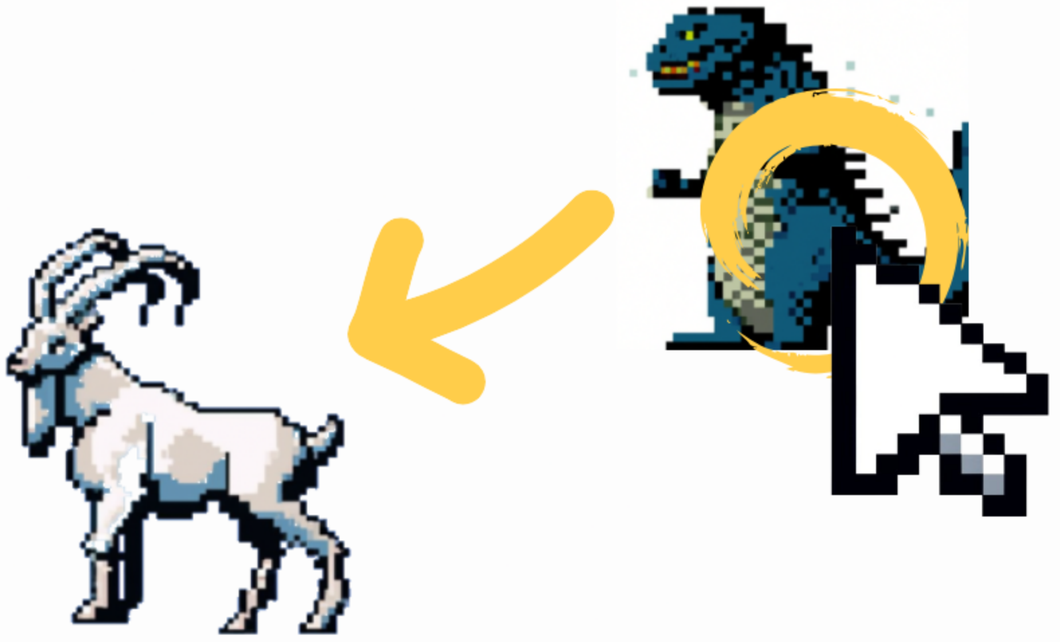

산

극지방

사용자 상호작용

추가 먹이 배급

드래그 앤 드롭으로 인위적 이벤트 생성



실제 BioNet 플레이

향후 BioNet 발전 방향

기존 구상은...

감사합니다.