[ 게임 데이터베이스 프로그래밍 ]

**게임 데이터베이스 생성 및 관리**

인하대학교 미래인재개발원 문화콘텐츠프로그래밍

김예슬

**1) 설계된 데이터 구조**

- 게임 맵 툴에 적용될 데이터 구조

우상단 / 좌상단 / 우하단 / 좌하단 위치 값

각 타일의 속성 값

플레이어 리젠 위치의 경우 1번 받아와야 함

각각의 값은 특정 알파벳 단어로 암호화가 되어 있어야 하고, 복호화를 통해 원래 값을 알 수 있어야 함

**2) 프로그램 소스 코드**

void Map::ReadData(HWND hWnd, TCHAR \*OpenFileName)

{

string readDataFile;

ifstream mapFile;

std::wstring wStr = OpenFileName;

readDataFile = string(wStr.begin(), wStr.end());

// >> TCHAR -> string

char \*changeFile;

changeFile = strtok(&readDataFile[0], "\\");

while (1)

{

changeFile= strtok(NULL, "\\");

if (changeFile == NULL)

{

MessageBox(hWnd, L"해당 파일을 지원하지 않습니다."

, L"파일 열기 실패", MB\_OKCANCEL);

return;

}

if (strstr(changeFile, ".dat"))

break;

}

mapFile.open(changeFile, ios::in | ios::binary);

if (mapFile.is\_open())

{

tileMap.clear();

// >> 새로 읽어오기 전 초기화

POINT pos = { 0,0 };

TileMap temp;

temp.pos = { 0,0,0,0 };

temp.type = 0;

// >> 플레이어

mapFile.read((char\*)&pos.x , sizeof(int));

mapFile.read((char\*)&pos.y, sizeof(int));

pos.x = (pos.x / dKeyCode) - dKeyCode;

pos.y = (pos.y / dKeyCode) - dKeyCode;

temp.pos.left = pos.x - 8;

temp.pos.top = pos.y - 8;

temp.pos.right = pos.x + 8;

temp.pos.bottom = pos.y + 8;

temp.type = ePlayerResen;

temp.showPos = SetShowType(temp.type);

tileMap.push\_back(temp);

// >> 맵 정보

while (!mapFile.eof())

{

mapFile.read((char\*)&temp.type, sizeof(int));

mapFile.read((char\*)&temp.pos.left, sizeof(int));

mapFile.read((char\*)&temp.pos.top, sizeof(int));

mapFile.read((char\*)&temp.pos.right, sizeof(int));

mapFile.read((char\*)&temp.pos.bottom, sizeof(int));

temp.type = (temp.type / dKeyCode) - dKeyCode;

temp.pos.left = (temp.pos.left / dKeyCode) - dKeyCode;

temp.pos.top = (temp.pos.top / dKeyCode) - dKeyCode;

temp.pos.right = (temp.pos.right / dKeyCode) - dKeyCode;

temp.pos.bottom = (temp.pos.bottom / dKeyCode) - dKeyCode;

if (temp.type < 0)

break; // >> 무한루프 처리

temp.showPos = SetShowType(temp.type);

tileMap.push\_back(temp);

}

}

mapFile.close();

}