2017 교내 공모전

상금 : 60/40/20

**심사 기준 : 창의성, 기능성, 상업적 가치성, 프로그램 구현 및 완성도**

6/5~6/30 : 신청접수  
6/ 5 ~ 8/25 : 작품 제작 기간  
8/25:  마감  
8/28 ~ 9/1 中 하루(추후공지) : 프레젠테이션(작품에 관한 설명) 및 교수님 심사  
9/4(예정 및 추후공지) : 개강총회(결과 발표 및 시상)

\* 프레젠테이션 : 완성되어 제출한 프로그램에 관한 특징, 기능 등을 설명하고, 제작하면서 겪은 어려움이나 그에 따른 해결 방법 등을 심사위원 분들에게 설명하는 자리 마련

**퀘스트 기반으로 한 생활습관 개선 게임**

* 자신이 습관으로 들이고 싶은 것 + 어플에서 제공하는 기본 퀘스트를 매일 수행
* 일정관리를 해주는 앱인데, 이 결과가 캐릭터의 성장에 영향을 미치는 일정 관리 앱
* 캐릭터를 마이룸에서 키우고, 일정을 성공적으로 수행시 게임머니 지급
* 캐릭터가 생활하는데 당연히 게임머니가 소비됨(생활비 등등)
* 일정을 수행하지 못하면 점점 피폐해지는 캐릭터를 보고, 양심의 가책을 느낄 것임  
  너무 피폐해지면 캐릭터는 사망하게 됨
* 게임 오버시, 새로운 캐릭터를 입양해 올 수 있다.  
  혹은, 성공적으로 계획을 일정기간동안(7일 or 14일 or 30일) 수행했을 경우, 캐릭터를 모두 성장시키고 새로 입양해 올 수 있다.
* 캐릭터를 죽이거나, 성장시킨 것 여부에 따라 게이머에게 칭호가 부여된다. 이 칭호는 구글 게임 플레이와 연동할 예정.  
  ex) 이제 막 습관을 들인 자, 이 앱을 왜 쓰는지 모를 정도로 성실하네요!, 마이너스의 손, 정말로 게으른 자, 그랜드슬램, 푸른수염 등등

**사용 프로그램**

* UI : Adobe Photoshop, Adobe Illustraitor
* 구현 : Unity, SQLite

**활동 내역**

* 아래에 시간 순으로 결정사항들을 날짜별로 나열
* 6.19 : 앱 아이디어 발상
* 6.20 : 앱 기획
* 6.22 : 공모전 신청서 제출
* 6.26 : 구현을 위한 게임 UI/UX 설계
* 6.27 : 개발툴을 Unity로 결정 및 구현 시작
* 6.28 : 레이아웃 구현  
  슬기 – 메인UI / 왼쪽슬라이드  
  명현 – 달력UI / 스크롤 구현 시도
* 6.29 : 레이아웃 구현  
  화면 배율 문제 해결, 스크롤 및 리스트 동적생성 구현 시도
* (6.30~7.1은 MT일정으로 X)
* 7.2 : UI 구현 어느정도 마무리 / 구조 설계 시작
* 리스트 동적생성 및 스크롤 구현 성공, UI 디자인 수정 완료
* 예제들을 분석하여 일반 파일 입출력, SQLite, MySQL 비교 🡪 SQLite 채택
* SQLite 간단 예제 공부 / 유니티 프로젝트에 SQLite 플러그인 추가
* 7.3 : 게임 구현을 위한 설계  
  게임 머니 지급 조건, 스탯 계산, 이벤트 처리 등등 설정
* 7.4 : 일정 관련 처리  
  db의 schedule table 파싱해서 배열에 저장 / 일정 배열을 토대로 목록 생성하기
* 7.5 : DB schedule관리 함수 작성 / 동적 리스트와 배열 연동 처리  
  객체나 배열을 통해 db관리하는 함수 작성
* 7.6 : DB USER\_DATA, CHARACTER 작성 / 달력 동적 구현  
  로딩 시 유저 정보를 db에서 불러와 초기화  
  오늘 날짜를 불러와 달력을 생성하며, 달 바꿀 때마다 달력 UI 업데이트
* 7.7 : 스크롤뷰 버튼 터치 오류 해결 / 일정 수행 시 게임머니 지급하기 구현
* 7.10 : 게임 접속 시, 당일 첫 접속인지 아닌지 체크 / 일정 추가 구현 및 디버깅
* 7.12 : 일정 삽입, 삭제 구현
* 7.13 : 리스트 좌표 에러 해결 / 게임 조건 이벤트 구현

**구현 전 생각하는 난관들**

* 동적인 크기의 리스트 구현(AndroidStudio에서 어댑터 패턴 느낌?) / 스크롤 추가
* 그 리스트들에 동적으로 날자 매기고, 정보 등록하기
* 유저 정보 저장(돈, 칭호, 컬렉션 수집상태 등
* 일정 저장 및 불러오기, 업데이트 (파일 입출력만 쓰나? 오라클, MySQL같은 DB를 쓰나? SQLite쓰나?)

**현재 스크립트 상태 / 게임 제외 (6월 29일~ 30일)**

* GM 스크립트 : UI Root에 있음. 싱글톤.
  + 달력 / 메인화면 / 슬라이드 메뉴 등등 오브젝트를 관리 (static)  
    → static들은 GM.myRoom 이런 식으로 참조 가능하다.  
    게임 시작 전에는 모두 object가 on 됨 → Awake()에서 변수 초기화 → Start()에서 myRoom빼고 모두 setActive(false)로 비활성화
  + 그 이외의 non static 멤버변수들은, GM.getInstance().XXXX 이렇게 참조 가능
  + LeftSlide On/Off함수 가짐
* MyRoom 스크립트  
  leftSlide obj가 myRoom obj보다 depth가 높다는 것을 기억하자! (콜리더 때문)  
  leftSlide 바깥쪽 터치시 메뉴 접히는 코드 가짐
* LeftSlide 스크립트  
  일정관리 버튼 누르면, 캘린더 화면으로 이동
* Calendar 스크립트

오른쪽버튼 누르면 마이룸으로 돌아가고 (다만 >화살표 버튼 안생김...이거 생기게 해야해)

왼쪽버튼누르면 touchCalendar화면으로 전환

* TouchCalendar 스크립트

오른쪽 버튼 누르면 캘린더화면으로 돌아가고

왼쪽버튼 누르면 추가하는 아이 위에 생김

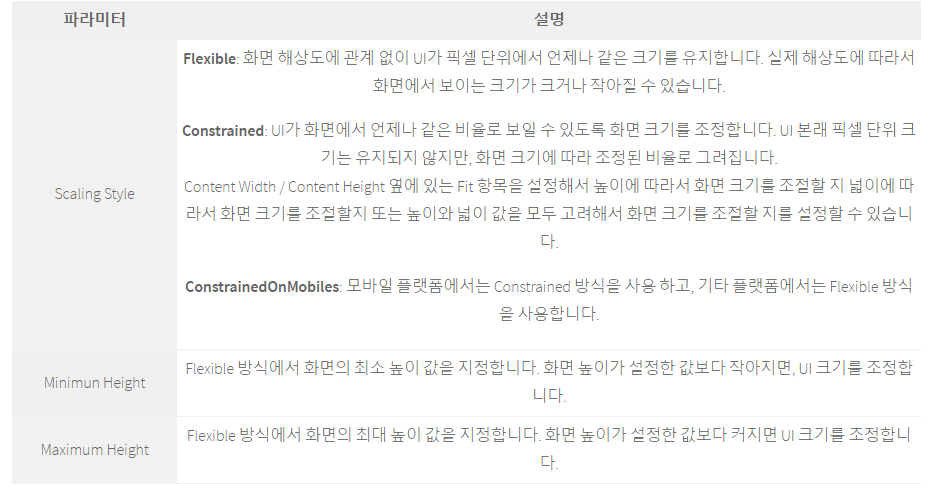
다만 얘의 문제점은 addwork함수안의 코드에서 SetActive(false)를 안해줘서 다이얼로그 하나 띄운거처럼 하려고 했는데 그거 안해주면 스크롤뷰가 다이얼로그보다 위쪽에 떠서 SetActive(false)를 해준다 비활성화 시켜도 보이는거보면 제대로 한것 같기도 하고 잘 모르겟다..

* AddWork 스크립트

오른쪽버튼(추가버튼)누르면 일정(스트라이프)추가 왼쪽버튼누르면 취소됨

**해상도 문제 (6월 29일)**

* <http://ronniej.sfuh.tk/ui-%EA%B8%B0%EB%B3%B8%EA%B5%AC%EC%A1%B0%EC%99%80-%EC%A3%BC%EC%9A%94-%EC%BB%B4%ED%8F%AC%EB%84%8C%ED%8A%B8/>



<http://www.devkorea.co.kr/bbs/board.php?bo_table=m03_qna&wr_id=42988>

멀티플랫폼이긴 하지만, ios상에다가 런칭할건 아니라서 720 \* 1280으로 해상도 변경 (potrait)

**UI 세부사항** **(6월 30일)**

: 동적 리스트 구현에 필요할 듯

* Left Slide UI
* 전체 크기 : 550 \* 1280
  + quest칸 사이즈 계산 방식  
    : 기본70 + 한칸당 65 \* item수 + 추가 여백 5
    - list item y 시작값 : -100
    - list item size : 550 \* 60
    - 다음 item y 위치값 계산 : -65 (item y size 60 + 여백 5)
  + slide : -640 → -90 // 가로 사이즈가 550
* Calendar
  + 날짜 한칸당 크기 90x90, 서로 100만큼씩 상하좌우로 떨어짐
  + 자세히 보는 칸은 700x425 위치 y:-415
  + 버튼들 50x50으로 맞춤
  + 위쪽 버튼들 위치는 x:+-250
* TouchCalendar
  + 여기도 마찬가지로 버튼 50x50으로 일단 맞춰둠
  + 위쪽 버튼 위치는 x;+-300
  + 스크롤바 안의 아이들도 대충 600x200으로 해둠
  + 스크롤바 크기는 600x900이고 위치는 y:350
* AddWork
  + 버튼크기 50x50 위치대강맞춰둠
  + user가 새로운 일정내용을 쓸 곳(inputDetail)은 550x300
  + 새로운 일정 이름 쓸곳(inputTitle)은 600x80 위치 Y:230
  + 돈 얼마로 등록할지 쓸 곳(inputMoney) 150x50 위치 x,y: -200

**일정 저장 방법 연구**  **(7월 2일)**

* 참고한 것 // 안쓰이는것도 쭉쭉 다 적은거라 이건 안봐도 됨
* <http://xx707xx.tistory.com/6> : DB는 MySQL사용 / 구현은 자바
* <http://mine7.tistory.com/64> : 자바 / 파일 3개이고, .dat파일 하나에 모두 저장  
  이 경우, 그냥 데이터 넣는 순서대로 id값을 매겼음(quto increment) / 탐색 시 모두 순회해야함
* <http://blog.naver.com/PostView.nhn?blogId=kosr81&logNo=110020295811> : C#, DB사용  
  이 코드를 볼 때, 블로그를 참고하자. 설명이 쓰여있다. / 문제점 : 변동사항이 DB에 저장이 안됨
* <http://freeprogram.tistory.com/29> : C#과 .NET사용 / 일정관리가 아니라, 그냥 달력임
* <http://yourpresence.tistory.com/87> 유니티 C# DB연동... 근데 SQLite가 아니라서 좀 아니다 이거.
* <http://dreamzelkova.tistory.com/entry/Unity%EC%99%80-SQLite%EC%97%B0%EB%8F%99%ED%95%98%EA%B8%B0> / Sqlite연동!!
* <http://blog.naver.com/PostView.nhn?blogId=algorithmlab&logNo=220877671743&categoryNo=0&parentCategoryNo=122&viewDate=&currentPage=1&postListTopCurrentPage=1&from=postView>
* <https://forum.unity3d.com/threads/tutorial-how-to-integrate-sqlite-in-c.192282/> 유니티 튜토리얼

**구조 설계 (7월 2일)**

* **일정관리 부분**
* SQLite를 이용할 것임  
  : 딱히 공부가 필요하지 않고, 간편해서.
* ex) scheduler.db에서, ‘Schedule’ Table의 형태  
  이거 그냥 예시임!!!! 계속 바뀔거야!!!!!!

|  |  |  |
| --- | --- | --- |
| Field Name | Type | 특이사항 |
| Date | TEXT | YYYY-MM-DD형태 |
| isEventOccured | BLOB | default : false / 이벤트 발생여부 |
| type | TEXT | BASIC, USER (defaule = USER) |
| isDone | BLOB | default : false / 일정 수행여부 |
| content | TEXT | 일정 상세 내용 |

업뎃 귀찮으니까 걍 나중에 DB캡쳐해서 올려야지~~

* **구현하는 부분 잡생각**  
  (단, 효율성 따위!)
* 삽입, 삭제, 탐색, 정렬 등 대부분 쿼리에 의해 쉽게 처리 가능할거임  
  ex) 일정 추가하는 경우, “2017년 10월 2일 친구 생일”이라는 일정을 추가했다고 가정하자.  
  그러면 INSERT into Schedule values (2017, 10, 2, 0, ‘친구 생일’);  
  <https://www.youtube.com/watch?v=BN1i4WvB8BY> 정 못하겠으면, 코드 작성시에 이거 참고
* 위의 이유로 인해, 굳이 db의 내용을 배열에 담지 않아도 괜찮을 것 같은데.
* 하지만 GM에 담아두는 것도 나름 장점 있지 않을까?  
  ex) Schedule mySchedule[item\_count]; //IDataReader, IDbConnection등 셋팅 다 했다고 가정  
  String detail = “TODO”; //다음에 order by 날짜순으로 해주면, 시간순으로 정렬됨  
  for(int i = 0; i< item\_count; i++){ mySchedule[i].detail = dbr.GetString(4); } 뭐 그런거?  
  🡺 직접 정의한 Schedule class배열에다가 쿼리 결과 저장하기!!  
  <http://www.androidside.com/bbs/board.php?bo_table=b49&wr_id=100824> 이런 느낌 ㅇㅇ
* 미리 GM에다가 배열로 담아두면, 쿼리문 덜 써도 될 듯  
  다만, 삽입/삭제/수정 등 db변동이 있을 경우, 재정렬 해줘야함(기준 : 날짜)
* 날짜를 기준으로 정렬하는 방법 : YYYY-MM-DD형태로 저장하고, 그냥 string 비교하면 됨  
  이때 주의할 점은, ‘3월 (X)’ 같은 형태가 아니라, ‘03월 (O)’ 이런 형태로 써줘야 한다는 것!
* **그러면 어떻게 구현할까?**
* 프로그램 실행 시, GM의 배열에다가 db를 날짜순 정렬해서 바로 저장
* Calendar에서 날짜 터치 시, ‘해당 날짜’ 기준으로 배열 탐색해서(혹은 쿼리수행) 결과 배열에 새로 저장  
  🡪 결과 배열을 토대로 동적 리스트 구현
* 일정 추가/삭제 시, ‘insert into Schedule (추가할 데이터);’와 같은 쿼리 수행 🡪 바뀐 DB를 다시 정렬 수행 🡪 GM의 배열에 다시 저장
* 일정 했다고 체크를 할 경우, Dialog 띄우자.  
  ‘해당 일정을 완료하시겠습니까?’  
  🡪 OK하면, db에서 해당 데이터의 BLOB값을 db 및 배열에서 수정 (이땐 굳이 배열 다시 안만들어도 됨. 값 하나 바꾸는거라서.)  
  🡪 OK한 이후에는 ~~이런 표시 해놓고~~ 해당 오브젝트 비활성화. 일정 한 걸 안했다고 푸는 걸 못하게 막자.

**게임(MyRoom) 설계 (7월 3일)**  
🡪 우선, 이 게임에서 모든 흐름의 중심은 ‘게임머니’

* 일정 수행시 게임머니를 지급함. 문제는, 어떻게 주느냐임.

ex) 어떤 사람은 사용자 등록 일정이 10개이고, 4개밖에 수행을 못함.  
근데 어떤 사람은 사용자 등록 일정이 2개이고, 2개 모두 완료했음.  
이때 수행한 %를 따져서 지급할건지, 성공적으로 완료한 일정 개수를 가지고 지급할건지, 일정마다 가중치를 둘 것인지 등을 생각해야함.  
단순하게 일정을 몇 개 완료했느냐로 게임머니를 지급하면, 쓸데 없는 일정 엄청 등록하고 몇 개만 실행해서 게임머니를 지급받을 가능성이 있기 때문임.

* 해결책 1 : 하루에 등록 가능한 일정 수를 제한한다. (최대 5개 이런식으로 ㅇㅇ)
* 해결책2 : 기본 퀘스트에 비중을 더 두고, 사용자 등록 일정에는 인센티브 수준으로만 제공한다.
* 해결책 3 : 수행한 %로 지급한다.
* **규칙 예시**
* 하루에 돈 받을 수 있는 일정의 최대치는 5개
* 기본 퀘스트는 매일 있으며, 각각 1000골드 지급  
  기본퀘스트는 설정에서 on/off 가능하다. 대신, off해두면 골드 못받는거니까 손해임 ㅎㅎ  
  ex) 매일 9시 기상, 물 3잔 마시기 등등
* 사용자 등록 퀘스트에다가 태그를 달고, 그에 따라 골드를 다르게 지급  
  ex.) 업무 및 과제 : 500골드, 약속 200골드, 자기가 등록한 매일 수행할 습관(ex도서관 가기) 300골드 등
* 골드 많은 순으로 정렬하고, 상위 5개만 합산
* 5개 합산골드 \* (수행일정/전체일정) = 지급할 게임 머니
* 게임 진행하는 동안, 이벤트 발생의 **기준 스탯 (MAX = 100)**  
  🡪 어떤 값들을 계산해야할지 생각해봐야함  
  🡪 이걸 조건으로 두고, 이벤트 발생 처리할거임  
   ex. 신뢰도가 낮으면 갑자기 아이가 범죄 저질러서 벌금내고 파산하고, 굶어죽음  
  🡪 망할거면 빠르게 망할 수 있도록 짜는 게 목표 / 난이도를 약간 높게 설정할 것임

🡺 현재 며칠 째 인지 정보, 전날의 스탯정보를 통해 평균 계산 가능함!!

* **실행력** : 매일 일정 수행 퍼센트의 평균을 계산한 값  
  = (수행한 총 일정 / 전체일정) \* 100 // 기본 퀘스트를 포함해서 계산
* **계획력** : 사용자 일정 개수의 평균 값 / 이 값과 실행력을 가지고, 신뢰도를 결정  
  = (일정 개수의 평균 값/ 5) \* 100 // 이 계산에서, 기본 퀘스트는 제외시킴  
  ex. 계획력은 높은데 실행력이 낮다 🡪 계획만 거창하고 수행 잘 안한다는 소리  
   계획력은 낮은데 실행력이 높다 🡪 한가한 사람이지만, 지키는건 잘 지킨다
* **건강** : 기본 퀘스트의 이행도  
  = (수행한 평균 기본퀘스트/전체 기본 퀘스트)\*100 //off해놨어도 계산에 포함. off면 안한걸로 처리됨~
* **IQ** : 랜덤하게 설정되는 값 / 불변
* **얼굴력** : 랜덤하게 설정되는 값 / 불변  
  못생기게 그려진 아이는 얼굴력이 낮을거임 ㅎㅎ
* **처리 순서**

1. bool isEventOccured : 전날의 결과에 따른 이벤트를 봤는지, 안 봤는지 플래그 / 일정 파일에 저장!!  
   날짜마다 달라지기 때문에, Scledule table에 넣어줘야 제대로 처리될 것으로 예상. default = false  
   🡪 해당 날짜에서, 앱을 처음 켰을 때만 false값이며, 아래 모든 것 처리 후 true로 바뀜.  
   중간에 게임을 종료해버린 경우, 그대로 false값이라서 처음부터 이 단계들을 실행하게됨.
2. isEventOccured = false일 때, 스탯에 따른 이벤트 발생 조건들을 체크함.  
   먼저 지금까지 평균 스탯을 계산한 후, 각 스탯들이 특정 값 이하인지 검사.  
   스탯이 모두 높으면 무사히 넘어가며, 여러 악조건을 만족시키는 경우는 ‘가장 나쁜 것’만 실행
3. 가장 나쁜 것이 실행되면서, 유저 화면에 출력됨.  
   (이때, 세금 등의 기본적으로 매일 나가는 돈도 따로 청구함. 지금 초점은 이게 아니라서 딱히 설명X)
4. 출력 이후, 결과에 따라 골드가 깎임. 골드를 깎은 이후, 파산 조건을 검사함.
5. 파산 조건 만족 시 아이는 바로 떠나며(세상을 떠나든가, 가출을 하든가 등등 엔딩), 새 아이를 입양  
   파산 조건 만족 안하면 축하합니다! 오늘을 버티셨습니다!

! **단, 이 처리에서 IQ, 얼굴력은 체크하지 않음.** 실행력/계획력/건강이 낮은지 체크해서, 함수를 호출할 것임.  
호출한 함수에서 IQ, 얼굴력 따라서 텍스트만 바꿀거임. (조폭이 될지, 호빠가 될지 이런 자잘한 텍스트)

! 아!! 이거 이전에 처리할거 하나 더 있어. 해당 날짜가 아닌, 전날의 isEventOccured를 체크하자!!!  
만약에 전날 접속을 안했으면, 아이는 바로 죽는걸로 하자!!

**( 단계 별 예시 )** // 간단하게 하기 위해, 건강만 보겠음  
위에 적은 단계와 숫자 동일함 ㅇㅇ

1. 오늘은 7월 3일 / 3일차 / 어제를 제외한 평균 건강력 : 50 / 어제 건강력은 20  
   이제 게임을 켜볼까? ㅎㅎ
2. 스탯 계산해야지. 오늘 평균 건강력 = (2 \* 50 + 1 \* 20)/3 = 40  
   건강관련 이벤트 발생 조건 : if(health <= 45)  
   45보다 낮으므로, healthPromblem(); 함수를 호출해.   
   (스탯이 여러 개라면, 뭐가 가장 나쁜지 우선순위에 따라 호출할 함수를 1개만 결정해야함)
3. [SYSTEM] 월세가 아닌 신개념 원룸! 일세! 3500골드가 차감됩니다. [ 확인 ]  
   [SYSTEM] 아이가 아픕니다. 치료를 하려면 약값이 듭니다. : 6000골드 [ 확인 ]
4. 9000골드 있었는데, 6000 + 3500골드 깎였다.  
   파산조건 : 정산 이후 이후 골드가 0 이하 / if(money <0)
5. 9000 – 9500 = -500 🡪 파산  
   방금 악조건 만족한 원인은 건강이었으니까, gameOver(); 함수 호출할 때 파라미터로 악조건을 넘기자.  
   (switch문같은거 써서 그에 맞게 텍스트 출력하면 되겠다!)  
   [SYSTEM] 아이가 아파서 운명했습니다. 생존 날짜 : 3일 [ 확인 ]

어떤 클래스 만들지 생각 대략 생각해두고, 역할 분담해서 짜면 되겠다!

**DB 및 관련 클래스 정보 (7월 4일)**

DB 설명

* db파일명 : scheduler.sqlite  
  StreamingAssets 폴더 내부에 있음
* db의 테이블  
  (7.4 기준 아직 한 개 뿐이지만, 나중에 유저 정보, 캐릭터 정보 등을 저장할거라 늘어날 예정)
  + schedule  
    용도 : 일정 저장

|  |  |  |
| --- | --- | --- |
| 필드 | 타입 | 특이사항 |
| content | TEXT | 일정 상세 내용 |
| type | TEXT | BASIC, USER (defaule = USER) |
| Date | TEXT | YYYY-MM-DD형태 ex) 2017-03-05 (2017-3-5 안됨) |
| isEventOccured | BLOB | default : false / 이벤트 발생여부 |
| isDone | BLOB | default : false / 일정 수행여부 |
| Classification | INTEGER | 0~4, 분류 / 골드 지급 다름 |

클래스 설명

* Schedule  
  : db의 schedule 테이블 정보를 담기 위한 클래스  
   테이블 필드 순서와, 생성자 parameter 순서랑 똑같음
* 구현한 것  
  **(7월 4일)**기본적인 setter, getter, print 함수

**(7월 5일)**SQL을 위해 bool type변수를 int형으로 get하는 함수 추가  
**(7월 7일)**enum Classification, getCost() 추가 (type과 classification 정보를 통해 게임머니 계산)

* LoadSaveManager  
  : DB 조작을 위해 만든 클래스
* 구현한 것  
  **(7월 4일)**  
  schedule 테이블 전체 일정을 List<Schedule>에 담아 리턴

**(7월 5일)**  
dbSetting함수 작성  
날짜를 parameter로 넘기면, 해당 날짜의 레코드들만 List<Schedule>로 리턴하는 함수 작성  
일정 수정/추가/삭제해서 db파일 갱신하는 함수 작성

**(7월 6일)**  
UserData와 CharacterData를 초기화하는 함수 작성  
날짜의 UserQuest와 BasicQuest 리턴하는 함수 분리

* UserData
* Character

**현재 게임부분 구현상태 (7월 14일)**

: **GM 123째줄~ 끝까지가 게임부분 관련 함수  
스크립트에 있는 요약글 + 함수 보고 대략적인 흐름을 이해해두세염!!!!**

* 오늘 접속 여부 따져서(checkToday함수), 첫 실행인 경우에만 이벤트 발생시킴  
  (추후에 다른 루트로도 발생하도록 추가할 수는 있지만, 일단 거른다)
* 기본 이벤트 발생 / 스탯 계산 / (스탯에 따른)조건 이벤트 발생 구현함
* 끝까지 안보고 종료하면, userData 바뀐 것이 DB에 저장 안되게 했음  
  따라서, 중간에 꺼버린 경우에는 다시 켜면 이벤트 처음부터 볼 수 있음  
  (다만, 이건 나중에 엔딩 볼 때 IEnumerator쓴다면 코드수정 있을 수 있음)

**혹시 헷갈릴까봐 칭호, 아이템UI, 아이템 리스트 정의**

* 칭호  
  : USER\_DATA의 killCount와 successCount에 따라 지급됨  
   메인화면에서 아이콘 터치 시, 획득/미획득 칭호 리스트가 나와야함 (아마도 list view)  
  ex) 앗! 실수!, 마이너스의 손, 게으른 자, 그랜드슬램, 푸른수염, 성실한 자, 이 앱을 왜 쓰는지 모를만큼 성실하네요 등
* 아이템 리스트  
  : 성공 엔딩 시 지급하는 아이템의 리스트  
  메인화면에서 아이콘 터치 시, 획득/미획득 아이템 리스트가 나와야함 (아마도 grid view)  
  ex) 캐릭터 코스튬, 장착 아이템(선글라스, 옷, 머리 리본 등), 방안에 둘 장식품, 가구 등
* 아이템UI  
  : myRoom에서 가구를 배치하거나, 캐릭터에게 아이템을 장착하려면, 보유 아이템들 적용 상태를 on/off 할 수 있어야함. 가구 위치 조작은 너무 어려우니까 일단 거르고, on/off 구현할 수 있는 UI가 필요함  
  ex) 머리 리본은 갖고있지만, 현재 캐릭터한테 안 입힐거라서 off, 가구 아이템은 on해서 룸에서 보이기

앞으로 우리가 해야할 일

**일정부분**

* 일정 수정 구현  
  (그냥 지원하지 말까? 안해도 될 듯 한데ㅇㅇ DB함수는 있어염)
* classification 선택 리스트 on/off시 잔상이 남는 에러  
  ( SetActive(false)하고 이것저것 해봤는데 해결 결국 못했어염..)  
  [Touch] 누르면 목록이 항상 열리고, 그 중에 아무거나 누르면 목록이 닫히도록 구현했으면 좋겠어염
* 자잘한 에러? (발견 못함)

**게임부분**  
**: 할 일이 많아서, 해야하는 순서대로 적었음**

* 배드엔딩 / 아이 성장 성공 엔딩 / 새 아이 입양씬  
  이렇게 3가지 UI를 어떻게 보여줄건지에 대한 시각적인 구상 & 구현방법 설계  
  **(이거 진짜 중요함)**
* 5가지 캐릭터 리소스 제작 (그림노동)
* 위의 3가지 씬에서 필요한 리소스 제작 (그림 노동)
* 3가지 씬 UI구현 / 처리 스크립트
* 아이템 컬렉션, 칭호 컬렉션, 아이템UI DB 구축 및 관련 함수(쿼리문 수행) 작성
* 아이템 리소스 제작 (그림노동)
* 컬렉션, 칭호 보여주는 리스트(아마도 grid&list view형태) 제작
* 엔딩 시 칭호, 아이템 지급 조건 검사하고, db업뎃  
  게임 접속 시 DB에서 불러와 자동으로 리스트 채우는 함수 구현
* 아이템UI에서 on/off 구현  
  아이템 on/off 상태도 저장 필수. 마찬가지로, 게임 접속 시 DB에서 불러와 자동으로 on/off 셋팅해야함
* 시간이 남는다면, google play game과 칭호 연동 작업

**※ 리소스 제작 관련**

일러스트레이터 사용하면 마우스로도 깔끔하게 얼마든지 만들 수 있음ㅇㅇ 타블렛 필요 X (안사도 됨)  
(계란이같은것도 걍 마우스로 만든거에여)

근데 툴 적응땜에 아마 내가 하는 게 빠르긴 한데, 혹시 디자인 양이 많아지면 그때 분업합시다 ㄱㄱ  
저장할 때 반드시 PNG-24로 저장하는 것 잊지 말기(PNG-8은 경계선 화질 깨짐, jpg는 투명도 손실)

**그래서 결론이? 이제 뭘 해야한다구?**

* 이번주 내로, 위에 써둔 것 일정부분 마무리
* 위에서 씬3개 구상 부탁해염ㅇㅇ 전에 달력, 게임UI짜던 방식대로!!  
  그동안에 내가 캐릭터 그리면서 같이 구상해볼게

bad/good 엔딩 발생 🡪 새 아이 입양씬 순서임 ㅇㅇ (당연한거지만)

씬을 구상하자.

* 배드 엔딩
* 아이 시점에서 서술하기
* 죄책감을 자극하고싶다
* 아사, 병사, 가출해서 장기적출 or 원양어선
* 성장 엔딩
* 까만 화면 : 아가를 찾음. 근데 없음
* 우체통으로 감. 편지가 있음
* 편지내용 : “고마워!”
* [획득] : 아이의 마음이 담긴 자그마한 선물상자

🡪 상자 오픈 씬 🡪 랜덤하게 얻기

* 입양씬
* 띵동- 택배왔습니다
* “? 뭐지?”
* 나가니까 상자가 있음. 열어보니 아가
* 세계관 적어놓는 아이콘 추가하기