|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 科目名 | 年度 | レポート番号 | クラス | 学籍番号 | 名前 |
| API実習 | 2023 | 5 | B | 20122077 | ROGER MARVIN |

ページ数や文字数よりも、読んでわかりやすく書けているかどうかが、点数アップの分かれ目です。  
APIを使ったアプリやゲームが作ったけど「動きませんでした、完成しませんでした」は評価に値しません。 単位取得は、きちんと動くものが評価対象です。 APIを使うこと、そしてプログラミングは１年生からの講義で学ぶことをすべて活用すれば実現できるはずです。

# 設問(1)

この科目で学んだ内容を第3者(他学部の学生や親など)にわかるように説明せよ。

この科目ではAPIについて学びました。APIというのはデータべース（.db, .sqlite, .csvなど）からデータをとって、あるエンドポイント（HTML、ゲームなど）に出すことです。仕組みは簡単にいうと、エンドポイント（例えばHTML）から何かをリクエストして、APIがそこリクエストに基づいてデータベースから必要なデータを読み込んで、JSONとしてレスポンスします。そのJSON形データが複数な人につかっるものなります。

APIでホームページに表示することです。まず、データをリクエストして、JSONのレスポンスがもらいます。そこで、データを綺麗に表示するためには、そのJSONをJavascriptなどでHTML形にして、HTMLのページに表示することです。例えば、あるWEBサイトにログインする時には入力したメールとパスワードをAPIでデータベースから同じのメールのパスワードを取って、入力したパスワードと比較します。もしも、正しければ、ログイン成功で、次のページ移動できます。もしも、間違ったら、404のページに移動するか、再入力してくださいのメッセージが出すことです。というわけで、APIはアプリやサービスのシステムを管理するためのものです。

# 設問(2)

レポート(4)をもとに、API連携作成またはAPIを用いたサービス開発結果を書いてください。何かしら動くものが出来ている前提です。

## 名称

Deadly Death Deadliest / デッドリー デス デッドリアス / 「デデデ」

## 概要(作ったものの説明)

WEBゲームです。このゲームはシングル　プレイヤー　ローグライク　スピードラン　オクション　スラッシュ　ゲームです。このゲームをやる目的はゲームは最短時間でゲームをクリアしながら、一番高いスコアを狙うことです。ゲームをクリアする度に、もしも、スコア一番高いとクリア時間一番早いだったら、ゲームの公式ウエブサイトにー自分の成果を表示します。  
このゲームをアクセスためにはゲームの公式サイトにアクセスできます。そこで、自分または他のプレイヤーの達成を見え

## サービス説明(動作がわかるように画面を交えて説明すること)

このWEBゲームは、ゲームの一覧を見えるし、プレイヤーがゲームをクリアしたプレイヤー名やスコアなどを他のユーザーに公開できて、TOPプレイヤーを10人までに表示できます。また、このゲームを1日何人ゲームをやるのかを計算して、表とグラフに出せます。  
実際のものは以下の通りです。　　  
A screenshot of a video game

Description automatically generated  
A screenshot of a game

Description automatically generatedA screen shot of a graph

Description automatically generated

こちらの部分はゲームのイントロダックションの部分です。ここでこのゲームはどんなゲームなのか、ゲームのストリーについてのものです。簡単に言うとあらすじです。

この部分はゲームを実際にやれるところです。この「PLAY」のボタンを押せば、HTML５式ゲームをやれます。  
その下の部分は「Leaderboard」の部分です。この部分では、ゲームクリアするのプレイヤー名やスコアの表とこのゲームをやっている人数が1日に何人いるのかの表です。

この部分はゲームをやっている人数の表のデータをグラフ式に表示するものです。X軸は人数で、Y軸は日付です。  
このグラフと表は十日間のデータを表示することです。

A screenshot of a computer

Description automatically generated

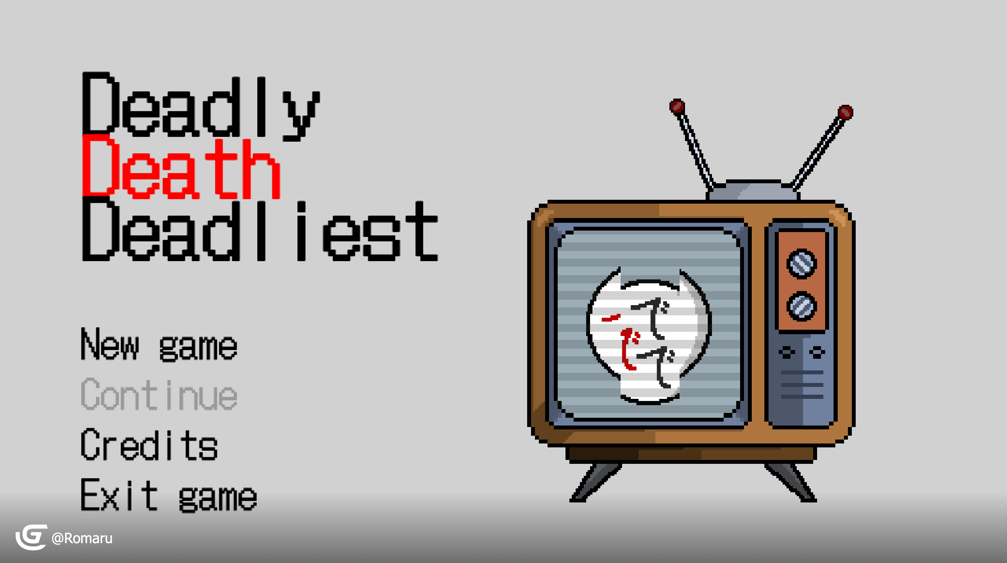
この部分は連絡の部分です。ここで、個人のメールアドレスとメッセージを送られ、開発者または管理者に直接にメールのやり取りができます。

ここはE mail.js APIを利用します。

A screenshot of a game

Description automatically generated

この「PLAY」のボタンを押せば、ゲームをできるようになります。

****

この「PLAY」のボタンを押せば、ゲームをできるようになります。

**ゲームの説明**

このゲームはシングルプレイヤーゲームで、パソコンしかできないゲームです。このゲームをアクセスために上のウエブサイトからアクセスできます。  
このゲームをパソコンのゲームで、ゲームの操作はキーボードとマウスで動かせます。

**A screenshot of a computer

Description automatically generated**

* 左の画像のように、キャラの動きは「ASWD」で動けます。
* ゲーム内の行為は「E」でできる。  
  ＊行為→チェスト開ける；ドア開ける
* 回復アイテムを利用するのは「F」でできる
* 攻撃のアクションは「マウス左」でできる

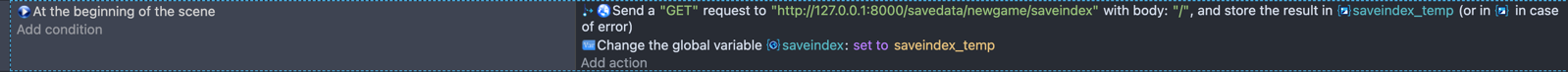
**A screenshot of a video game

Description automatically generated**

* こちらはゲームの画面です。
* 見た通りにHP、レベル、回復アイテムの個数、スコア、時間、メニューボタンとターゲットCursorが表示しています。これはどのステージでも、同じ形で表示します。

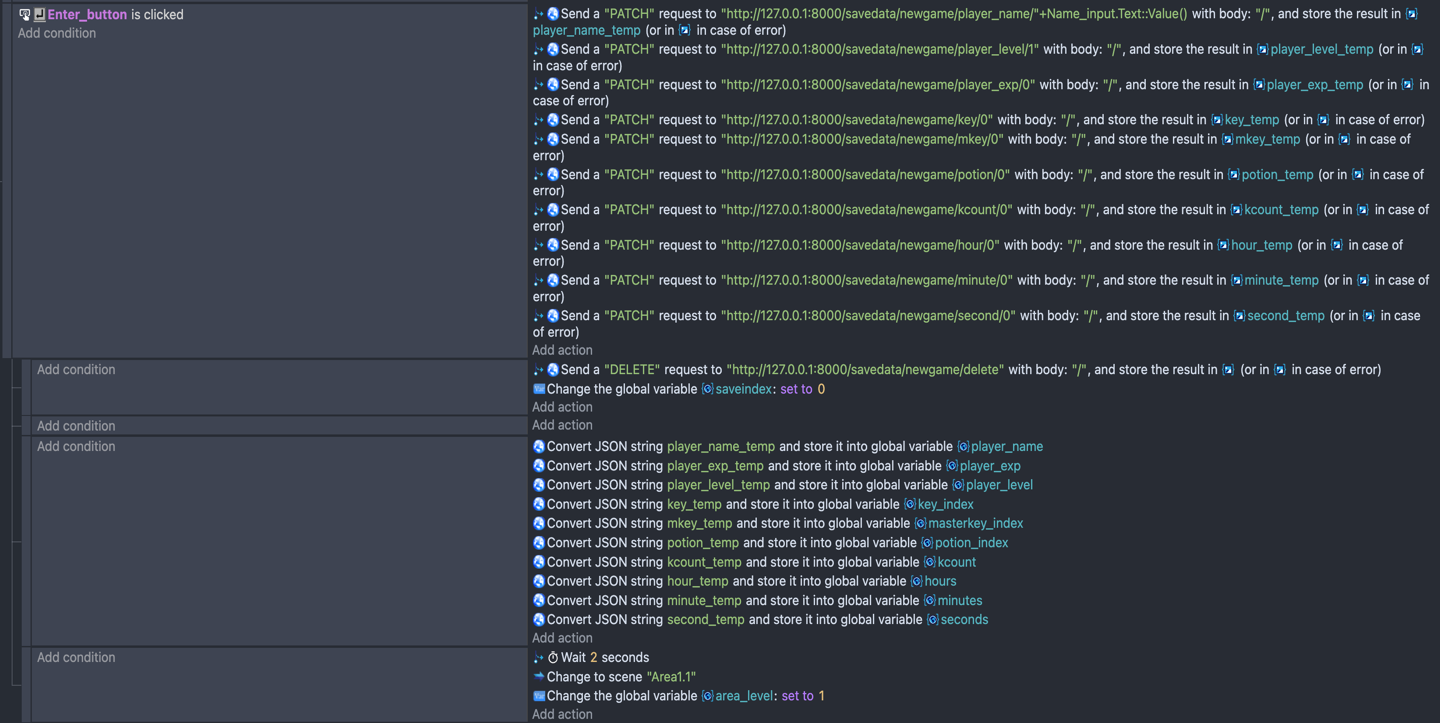
**ゲームとAPIの関連の説明（APIの動作）**

1. ゲームを開始する際に、ゲームがAPIにサーブデータを要求する。



これは、セーブデータが存在しているかどうかを確認するためです。もしも、Saveindexが０ではないという場合は、セーブデータが存在していると分かって、「Continue」のボタンがクリックできるとなります。

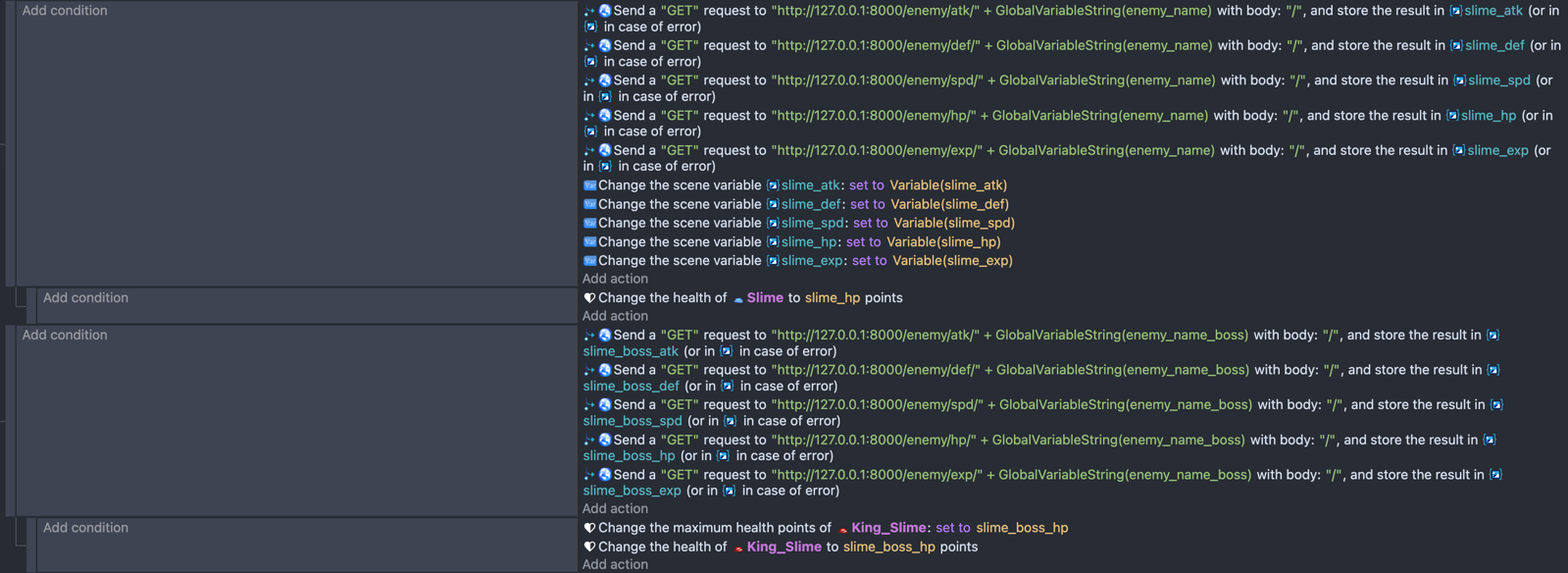
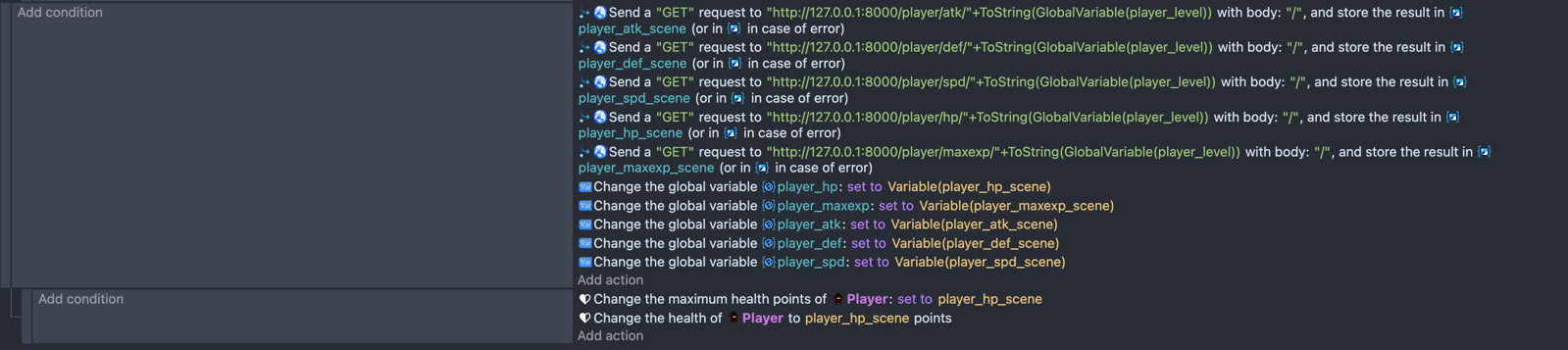
1. 「New Game」を選択したら、プレイヤーの名前を入力ボックスがあります。それを入力して、ゲーム内の「Enter」ボタンをしたら、ゲームが開始します。NewGameが開始する際に、ゲームはAPIに“PATCH”と“DELETE”をします。



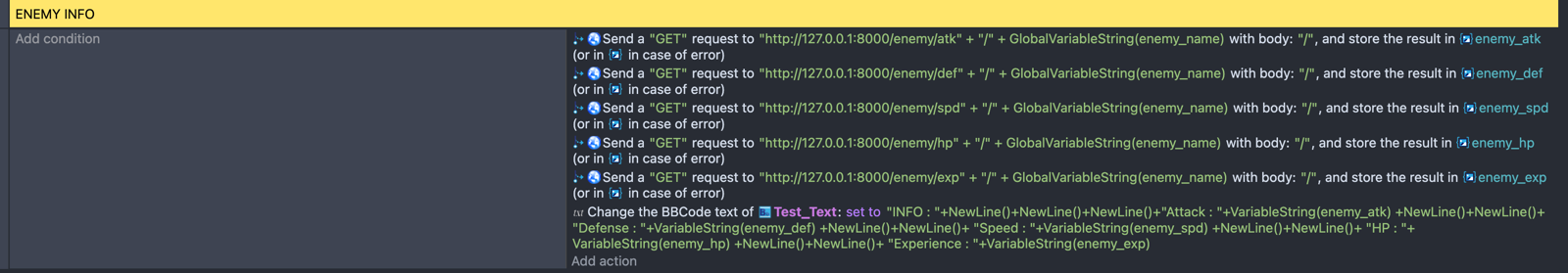
ここでSavedata.sqliteにあるのId=0の行をNewGameのデータに書き直すことです。そして、Id≠０の行を全部削除することです。Id＝０行のデータを書き直した後で、ゲームの変数に入れます。

**＊GdevelopのエンジンがAPIに要求する機能があって、要求したものをある変数に入れる機能が持っています。例えば、上のように、要求するのレスポンス(上から１行目のplayer\_name）をplayer\_name\_tempに入力することです。**

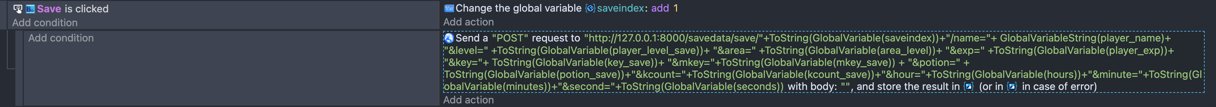
データベースから削除した後、Saveindexを０にセットして、tempの変数をグローバル変数（よく使う変数）に入れることです。全てがおわたら、ゲームのArea1.１に移動して、ゲームを始めます。

1. ゲームが始まる時に、ゲームシステムがAPIにプレイヤーと敵のステータス情報を要求する。  
   敵のデータ（ATK、DEF、SPD、HP、EXP）  
     
   プレイヤーのデータ（ATK、DEF、SPD、HP、MAX EXP）  
   敵の方で（enemy\_name）と（enemy\_name\_boss）の変数があって、それをこのステージに登場する敵の名前です。その変数で、敵のステータスをゲットできます。プレイヤーも同じですが、名前だけではなく、レベルでステータスを要求することとなります。レベルによる、ステータスが違います。  
   ステータスのデータをゲットした後、適切な変数に入力します。
2. **A screenshot of a video game

   Description automatically generated**「Almanac」を押せば、敵の図鑑を見える。  
     
   ここで、左の敵のリストの中のいずれかを押せば、敵の情報やイレストが表示します。  
   敵の名前を押すたびに、ゲームシステムがAPIに要求します。A computer code on a black background

   Description automatically generated

ここで、敵の名前を利用して、敵の情報を要求します。レスポンスをTest\_Textに構造して、表示します。

1. 「Save」を押せば、ゲームセーブできます。  
   セーブボタンを押す際に、ゲームシステムがAPIにPOSTの要求をします。  
   POSTする前に、saveindexを１プラスします。そして、データベースにデータを入れます。  
   このクエリでの説明：

"http://127.0.0.1:8000/savedata/save/"+ToString(GlobalVariable(saveindex))+"/name="+ GlobalVariableString(player\_name)+ "&level=" +ToString(GlobalVariable(player\_level\_save))+ "&area=" +ToString(GlobalVariable(area\_level))+ "&exp=" +ToString(GlobalVariable(player\_exp))+ "&key="+ ToString(GlobalVariable(key\_save))+ "&mkey="+ToString(GlobalVariable(mkey\_save)) + "&potion=" + ToString(GlobalVariable(potion\_save))+"&kcount="+ToString(GlobalVariable(kcount\_save))+"&hour="+ToString(GlobalVariable(hours))+"&minute="+ToString(GlobalVariable(minutes))+"&second="+ToString(GlobalVariable(seconds))

セーブする内容はプレイヤー名、プレイヤーレベル、エリアレベル、現在のEXP数、インベントリのものの個数（鍵、ボス鍵、Potion）、キルカウント（敵を倒した数字）、時間の数字（Hour,Minute,Second）

ここでプレイヤーレベルとインベントリアイテムはゲームを登場する時に別の変数にセーブした変数を使っています。このゲームのセーブメカニズムは、ステージに登場したばかりの状態をセーブします。もしも、ステージの途中までにセーブしても、ゲームを「Continue」する際に、ステージの途中から続くではなくて、ステージの最初から続くこととなります。

1. 時間の動作は下のようにです。  
   A screenshot of a computer

   Description automatically generated  
   ゲームが始まる時に、savedata.sqliteからデータを取って、ゲームに表示します。  
   もしも、hours/minutes/secondsが10<場合は、表示するときに”0”hour+”:0”+minute”:0”+seconds  
   10＞の場合は、hour+”:”+minute+”:”+secondです。  
   普通の時間のようにsecondが60>なら、secondsを０にセットして、minutes+1となる。Hourの場合も同じです。
2. Area1.3（現在のバージョン）をクリアできたら、ゲームシステムがAPIに“POST”の要求です。  
   A blurry image of a computer screen

   Description automatically generated

このクエリはleaderboard.sqliteのデータベースにプレイヤーの情をPOSTすることです。  
POSTするものはプレイヤー名、プレイヤーレベル、Kill Count、と時間です。  
データを投稿する前に、API先にそのプレイヤー名を確認します。もしも、存在している場合は、そのデータを書き直します。もしもなかったら、新しい投稿します。

＊しかし、投稿する際に条件があります。Kill Countとレベルの場合は、レベルとKill Countの数値が高いの方を投稿します。逆に、時間は数値が低いの方を投稿します。これで、プレイヤーがゲームをチャンレンジできて、一番多いキルカウントを目指していながら、短時間でゲームをクリアします。

**拡張性**

このWEBゲームの仕組みで、ゲームだけではなくて、色なサービスとして使えます。まず、このWEBゲームが使ったAPIがプレイヤーのスコアとゲームのログイン回数のデータを集取ことが一つの機能です。同じのような、仕組みで、例えば、教育用のWEBサイトが作れます。例えば、「PLAY」のボタンを押したら、ゲームの代わりに、テスト問題のアプリがでたら、その学生が問題を答えて、学生がよくできたか、あまりできないのかを集取したデータから自動分析できます。学生は自分の能力が見えるし、先生や教員などの担当の方々がデータを見て、この学生をどのように教えるのかをわかるようになります。  
教育以外は、政府がアンケートなどのデータ収集時でも、このサービスの仕組みで、データ収集が早めにできるし、解凍する方々もいいビジュアルで楽しみながら答えます。もちろん、その時は、個人情報が関係ある可能性があるので、セキュリティのシステムをつけることが必要です。

## レポート(4)の記載内容の実現状況（原則100%となること）

内容の実現状況8０％です。