

GENERAL OPERATIONS,

1. **ADDTOANDSAVE METHOD:** UPDATES THE VALUE OF A PROPERTY AT A SPECIFIED PATH IN FIREBASE REALTIME DATABASE BY ADDING THE GIVEN VALUE TO THE CURRENT VALUE OF THE PROPERTY.
2. **SUBTRACTFROMVALUEATPATH METHOD:** UPDATES THE VALUE OF A PROPERTY AT A SPECIFIED PATH IN FIREBASE REALTIME DATABASE BY SUBTRACTING THE GIVEN VALUE FROM THE CURRENT VALUE OF THE PROPERTY.
3. **UPDATEDATA METHOD:** SAVES DATA TO FIREBASE REALTIME DATABASE AT A SPECIFIED PATH AND VALUE NAME, WITH AN OPTION TO USE A RANDOM KEY.
4. **SAVECUSTOMCLASSTODB METHOD:** SAVES A CUSTOM CLASS OBJECT TO FIREBASE REALTIME DATABASE AT A SPECIFIED PATH, WITH AN OPTION TO USE A RANDOM KEY.
5. **LOGINANDREGISTER METHOD:** PERFORMS USER AUTHENTICATION USING FIREBASE AUTHENTICATION, WITH AN OPTION TO REGISTER A NEW USER.
6. **SENDPASSWORDRESETMAIL METHOD:** SENDS A PASSWORD RESET EMAIL TO THE SPECIFIED EMAIL ADDRESS USING FIREBASE AUTHENTICATION.
7. **SIGNOUT METHOD:** SIGNS OUT THE CURRENT USER FROM FIREBASE AUTHENTICATION.
8. **A SINGLETON CLASS THAT IS USED TO GET AN INSTANCE OF THE ADVANCEDDATABASE CLASS.**
9. **GETCUSTOMDATAWITHJSON:** ENTER THE FILE PATH AND DESERIALIZE THE INCOMING JSON DATA WITH JSONCONVERT.

ADDTOANDSAVE,

```
public void AddScoreToUser()
{
    //Here, we enter the path and the name of the variable, then we enter the value to be added at the
    end and call the function. After calling the function, it will automatically add the value to your
    variable in the database up to the amount specified in addValue.

    string path = "/users/john";
    string valueName = "points";
    int addValue = 20;

    //Call
    AdvancedDatabase.Instance.AddToAndSave(path,valueName,addValue);
}
```

SUBTRACTFROMVALUEATPATH,

```

public void Dead()
{
    //Updates the value of the given property at the specified path in Firebase Realtime Database by
    subtracting the given value from the current value

    string path = "/users/john";
    string valueName = "points";
    int subtractValue = 5;

    //Call
    AdvancedDatabase.Instance.SubtractFromValueAtPath(path,valueName,addValue);
}

```

UPDATEDATA,

```

public void UpdateUserPoints()
{
    //It changes the valueName with the value entered by going to the given path.

    string path = "/users/john";
    string valueName = "points";
    int setValue = 50;

    AdvancedDatabase.Instance.UpdateData(path,valueName,setValue);
}

```

SAVECUSTOMCLASSTODB,

```

public void SaveCustomClass()
{
    //saves a custom class object to Firebase Realtime Database at a specified path, with an option to
    use a random key

    Player myPlayer = new Player();
    myPlayer.coin = 30;
    myPlayer.name = 50;

    string path = "/users/"
    bool useRandomKey = false; //Like this /users/NKA-SD-234XS
    string customId = myPlayer.UID;
}

```

```
//Call  
AdvancedDatabase.Instance.SaveCustomClassToDB(path,userRandomKey,customId);  
}
```

AUTHENTICATION,

THERE IS A DEMO SCENE FOR THIS.

RESET PASSWORD,

```
public void ResetPassword()  
{  
    //Reset your password  
  
    string yourEmail = "test@test.com"  
  
    //Call  
    AdvancedFirebase.Instance.SendPasswordResetMail(yourEmail);  
}
```

GETCUSTOMDATAWITHJSON,


```
public void GetCustomData()  
{  
    //Enter the file path and deserialize the incoming JSON data with JsonConvert.;  
  
    string path = "/users/john";  
}
```

```
var strJson = AdvancedFirebase.Instance.GetCustomDataWithJson(path);

var player = JsonConvert.DeserializeObject<Player>(strJson);

//Now you have player class
Debug.Log(player.coin)
}
```

ENCRYPTION



```
public void CryptValue()
{
    string message = "Hello World!";
    string cryptPassword = "Hyperspace";

    //encrypt
    string encryptedMessage = AESCrypt.Encrypt(message, cryptPassword);

    //decrypt
    string decryptedMessage = AESCrypt.Decrypt(message, cryptPassword);

    Debug.Log($"Encrypted message {encryptedMessage} Decrypted message {decryptedMessage}")
}
```

THANKS FOR USING MY PACKAGE!