

How to Create Login Screen in PowerApps

- [Sign in the PowerApps](#) app with your credentials.
- Create a blank new Canvas app and choose any one Layout either **Tablet** or **Phone** layout.
- Next, it will open a Blank Powerapps screen. Mainly, We need two screens where one screen is for User Login (where the user can enter the name and password) and another one is to display the Welcome message.
- To display the Welcome message, We can add a new blank screen (**Home -> New screen -> Blank**) and rename it to **Welcome Screen**.
- In the Welcome screen, Just add two Labels and set its Text property to **"You Successfully Logged in!!"** and **"WELCOME TO POWERAPPS"** as below:
-

Step-2:

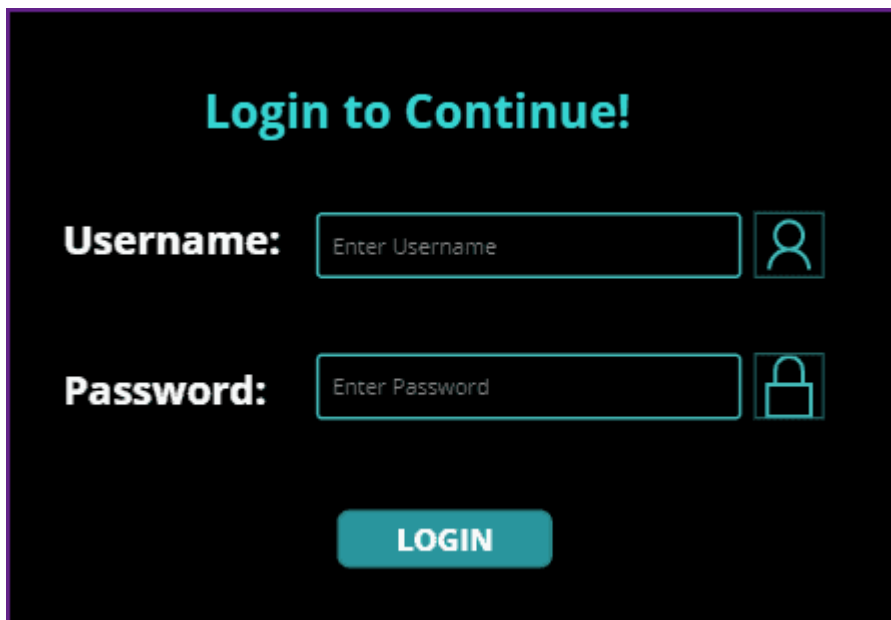
On the User Login screen, add all these below Powerapps input controls as:

- Add three **Label** controls (One is for Title and others are for the Username and Password).
- Insert two **Text input** controls (One is for Username and another one is for Password).
- Make two text input's **Default** property as blank and set both's **Hint text** to (Enter Username and Enter Password).
- Select the **Password** text input control and set its **Mode** property as **Password** or **TextMode.Password**.
- Insert one **Person Icon** (Insert -> Icons -> Person) and place it besides in the Username field.
- Insert another **Icon as Lock** (Insert -> Icons -> Lock) and place it besides in the Password field.

- Next, add one **Button** (Insert -> Button) and set its **Text** property to **LOGIN**.

After adding all the Powerapps input controls in the User Login screen, it will look like as the below screenshot:

After adding all the Powerapps input controls in the User Login screen, it will look like as the below screenshot:



Powerapps user login

screen

Step-3:

Select the **LOGIN** Button and apply this below formula on its **OnSelect** property as:

```
OnSelect = If(  
    txtPassword.Text = "TSInfo",  
    Navigate(  

```

```

        'Welcome Screen',
        ScreenTransition.None
    ),
    Notify(
        "Invalid Password!!! Please Enter a Valid Password",
        NotificationType.Error
    )
)

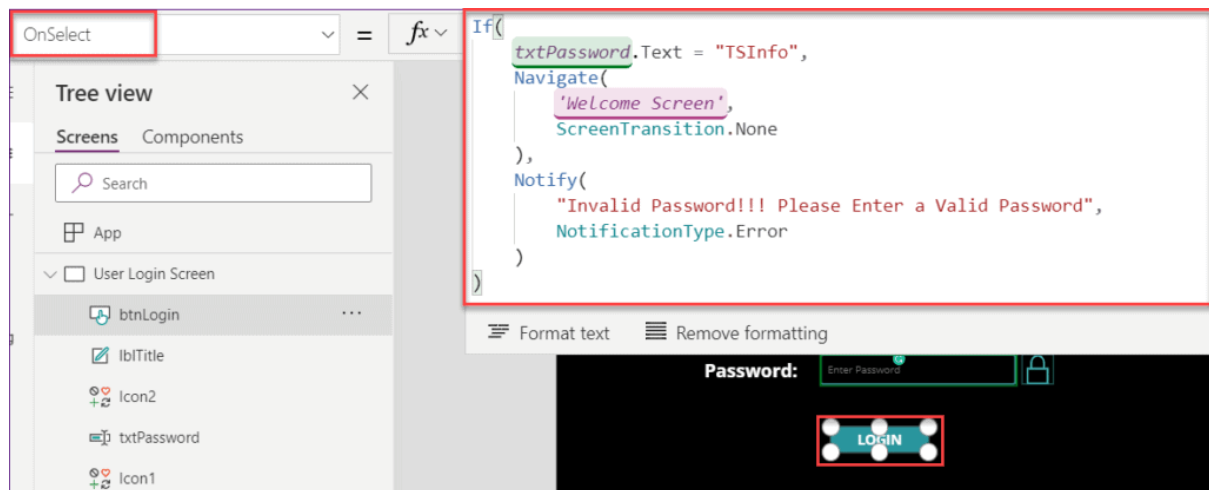
```

Where,

- **txtPassword** = Password text input control name
- **"TSInfo"** = I set the Password as TSInfo. Not only TSInfo but also you can set anything here inside the " ".
- **Navigate** = Navigate is the function that helps to change from one screen to another screen.
- **'Welcome Screen'** = This is the Screen name. When the user will successfully login without having any error, then this screen will appear.
- **ScreenTransition.None** = This is the transition argument where the new screen quickly replaces the current screen.
- **Notify** = Notify function helps to display a notification message to the end-user on the Power apps screen.
- **"Invalid Password!!! Please Enter a Valid Password"** = This is the error notification message that will display when a user will enter an invalid password.
- **NotificationType.Error** = This notification argument specifies the display message is an error message.

If you are interested to know more details about the Powerapps Notification function, then refer to this below link:

[How to use PowerApps notify function](#)

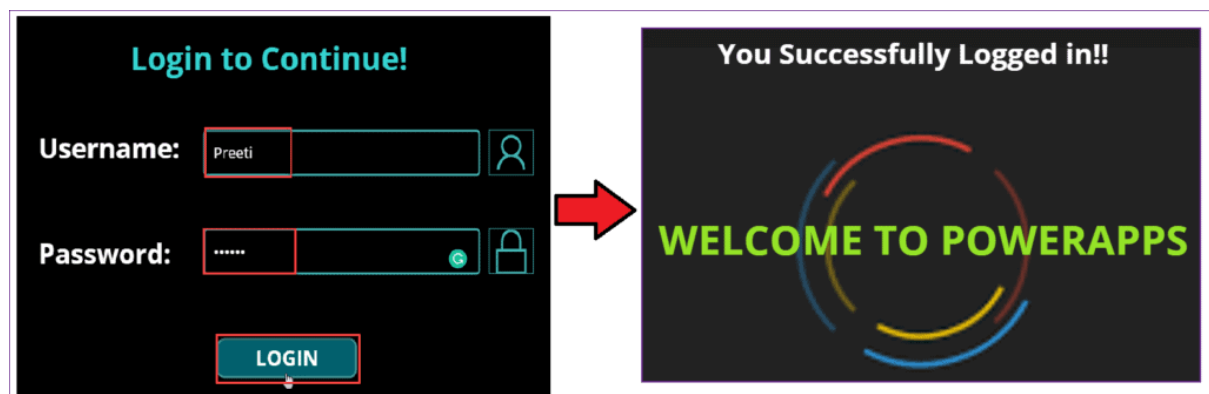


microsoft powerapps login screen

This above code specifies, if the user will enter the Password as “**TSInfo**”, then it will navigate to the Welcome screen otherwise it will show the notification error message.

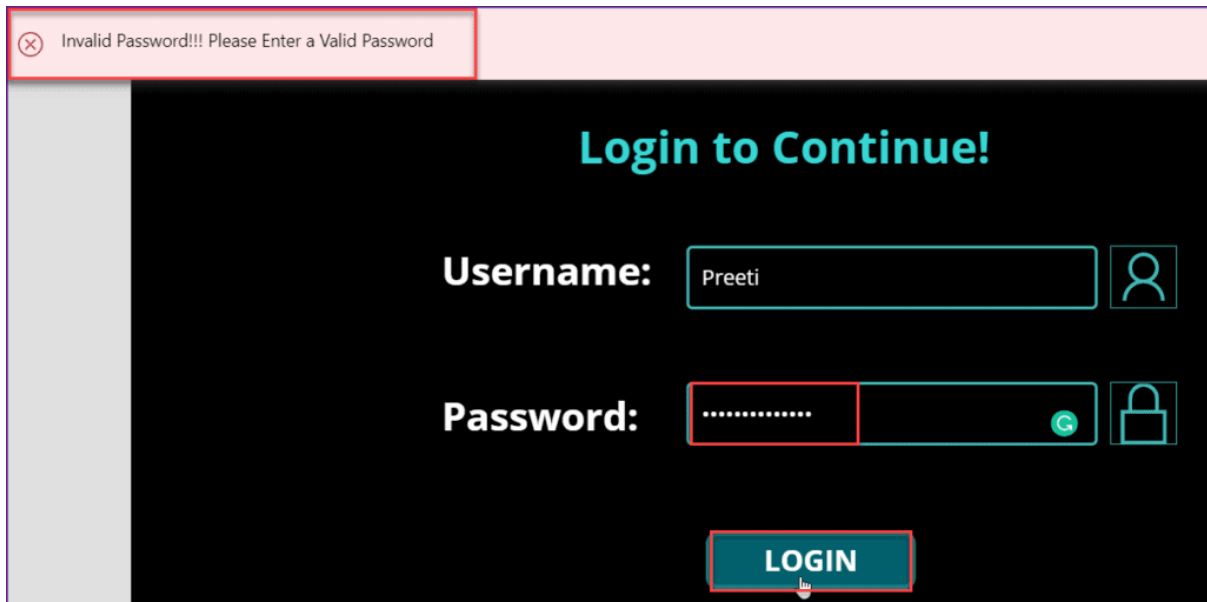
Step-4:

- Just save and preview (F5) the app. Enter the Username (any name) and Password as **TSInfo** and hit on the **LOGIN** button. Here, the Password input field is appearing with “.....”, because I set its Mode property to Password.
- If the Password validation is correct, then at the same time the Welcome screen will appear as shown below.



login screen design powerapps

- When you will enter an invalid password and then **LOGIN**, then the error notification will appear on the screen as like below:

A screenshot of a login screen titled "Login to Continue!". It features a dark background with white text. The "Username:" field contains the text "Preeti" and is accompanied by a user icon. The "Password:" field is masked with dots and is accompanied by a lock icon. A red error message banner at the top left reads "Invalid Password!!! Please Enter a Valid Password". A red box highlights the "LOGIN" button at the bottom.

powerapps login screen office 365

Create Powerapps Login Screen using SharePoint List

Here we will see how we can create Powerapps Login Page or screen using the [SharePoint List](#). Basically, we will use the SharePoint list to store the users' credentials.

When a user will log in to the [Powerapps app](#), it will check whether the specific user is existing or not from the [SharePoint List](#). If the user will exist, then the app will log in and navigate to the successful screen otherwise it will display with an error notification.

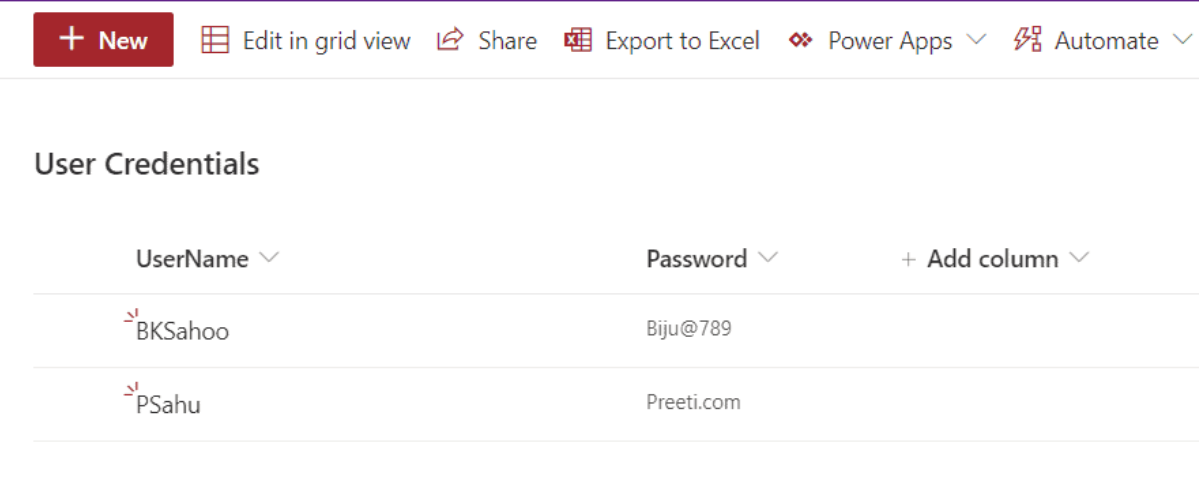
Follow these below steps to do so:

Step-1:

I have created a [SharePoint Online List](#) named as “**User Credentials**”. This list has two columns as:

- **Username:** By default, this is a Title column with a Single line of text Data type. Just I renamed it to **Username**.
- **Password:** This is a single line of text data type that I created.

After creating the columns, just add some new records into it. The below screenshot represents my SharePoint Online List (User Credentials):



User Credentials		
UserName	Password	+ Add column
BKSahoo	Biju@789	
PSahu	Preeti.com	

SharePoint Online List

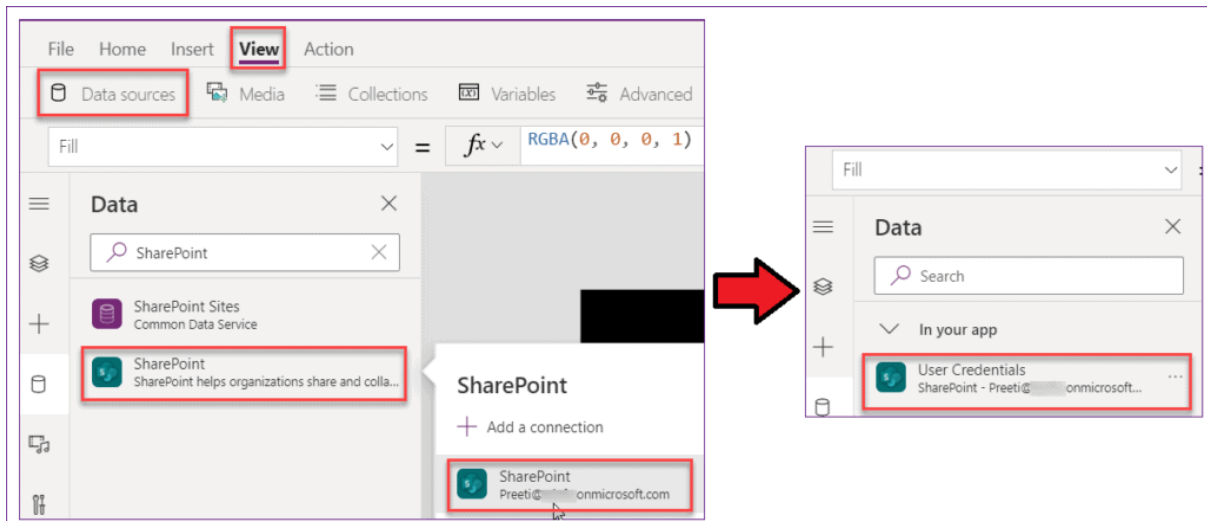
Step-2:

Next, Go to your [Powerapps app](#) and connect the SharePoint Data source to your app.

- Go to **View** tab -> **Data sources** -> Search **SharePoint** -> Add a new or existing connection -> Select your specific SharePoint

Site -> Choose the specific list (**User Credentials**) -> Hit on **Connect** button.

- Then your SharePoint Online list will be added to your app as like the below screenshot.



PowerApps connection

Step-3:

- Now, select the **Log in** Button from the app and apply this below formula on its **OnSelect** property as:

```
OnSelect = If(
    CountRows(
        Filter(
            'User Credentials',
            And(
                txtUsername.Text = Title,
                txtPassword.Text = Password
            )
        )
    ) = 1,
    Navigate(
        'Welcome Screen',
        None
    )
)
```

```

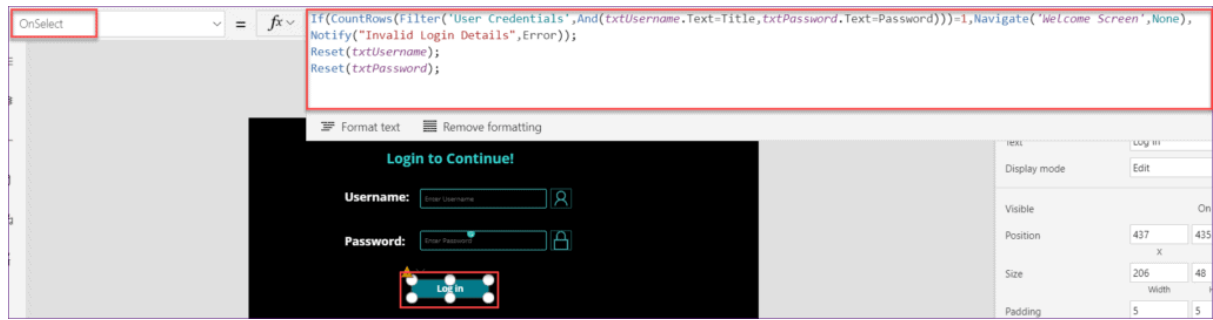
    ),
    Notify(
        "Invalid Login Details",
        NotificationType.Error
    )
);
Reset(txtUsername);

Reset(txtPassword);

```

Where,

- **'User Credentials'** = SharePoint Online List that contains the Username and Password of a user
- **txtUsername** = Username text input control name
- **txtPassword** = Password text input control name
- **Title** = SharePoint column name that I renamed to Username and it contains the username of a user
- **Password** = SharePoint column name that contains the password of a user
- **'Welcome Screen'** = This is the Powerapps Screen name. When the user will successfully login without having any error, then this screen will appear.
- **Notify** = Notify function helps to display a notification message to the end-user on the Power apps screen.
- **"Invalid Login Details"** = This is the error notification message that will display when a user will enter an invalid username or password.
- **NotificationType.Error** = This notification argument specifies the display message is an error message.
- **Reset(txtUsername)** = The Username field will clear once the user login.
- **Reset(txtPassword)** = The Password field will clear once the user login.

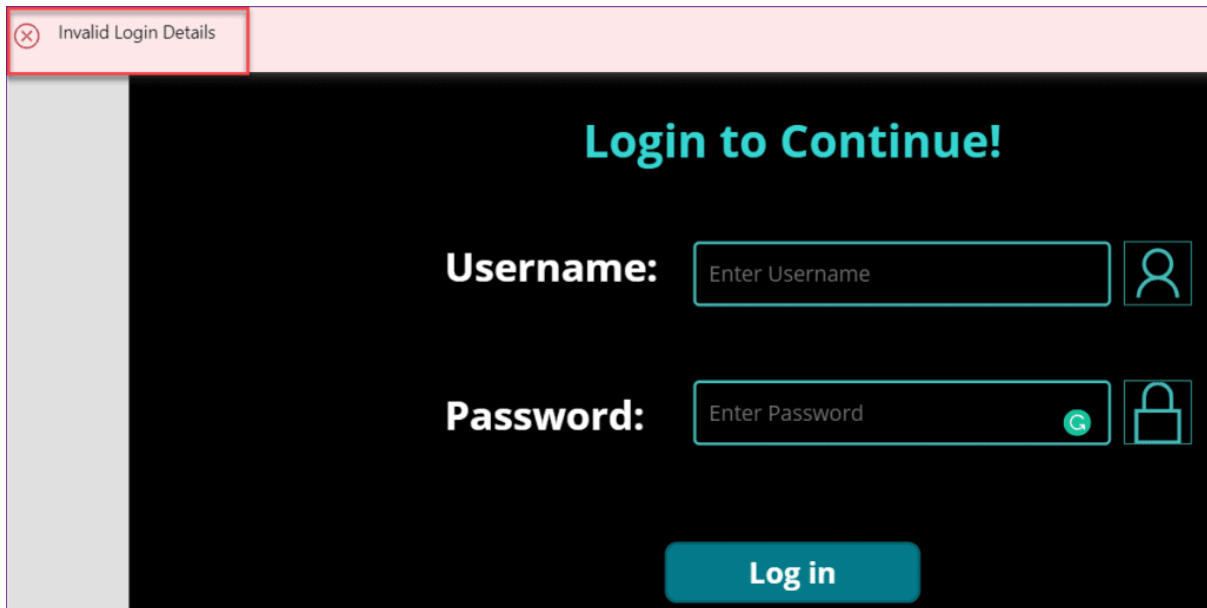


Step-4:

- Just save and preview (F5) the app. Enter the Username (any name) and Password of a user from the SharePoint list and hit on the **Log in** button.
- If the Username and Password validations are correct, then at the same time the Welcome screen will appear as shown below.



- When you will enter an invalid Username or an invalid password and then Log in, then the error notification will appear on the screen as like below:



PowerApps custom login screen

Create Login page in PowerApps Collection

Suppose, In Powerapps, you want to store some important User Login details. Then, in this case, you can collect it by using the [PowerApps Collection](#).

Here, all things are same as like above examples. Just we need to take a Button and apply the Powerapps Collection formula on its **OnSelect** property.

- On the Powerapps screen, Insert one **Button** named as “**Login to Collect**” and apply this below formula on its **OnSelect** property as:

```
OnSelect = Collect(  
    UserLogin,
```

```

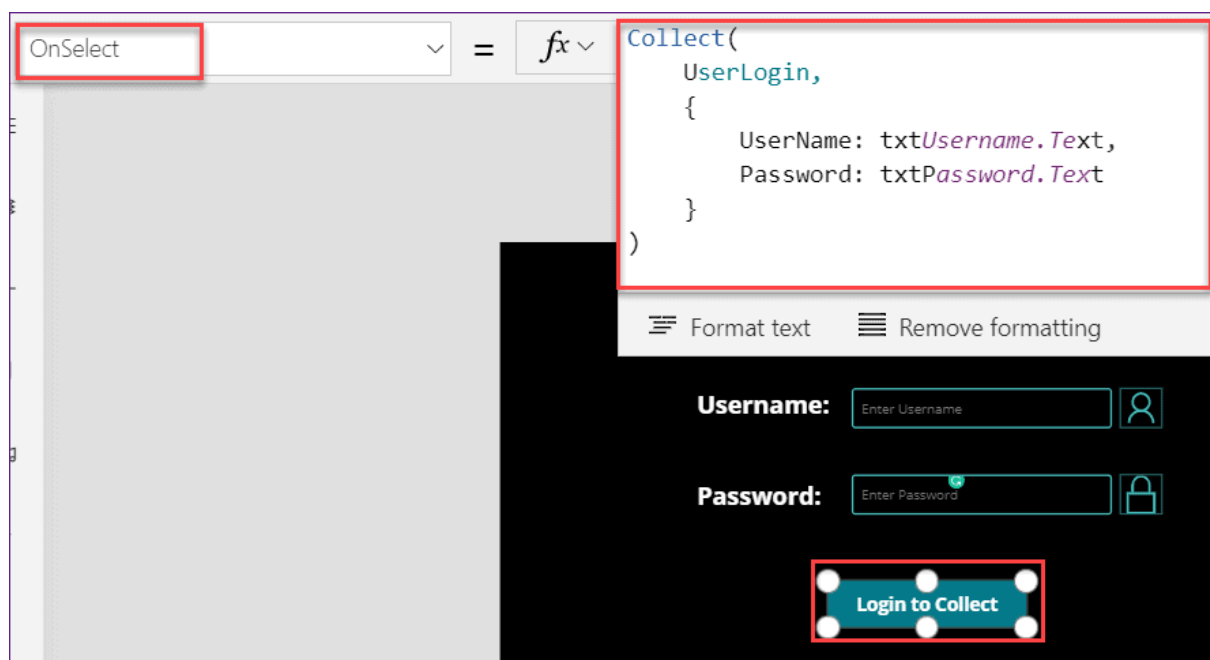
{
    UserName: txtUsername.Text,
    Password: txtPassword.Text
}

)

```

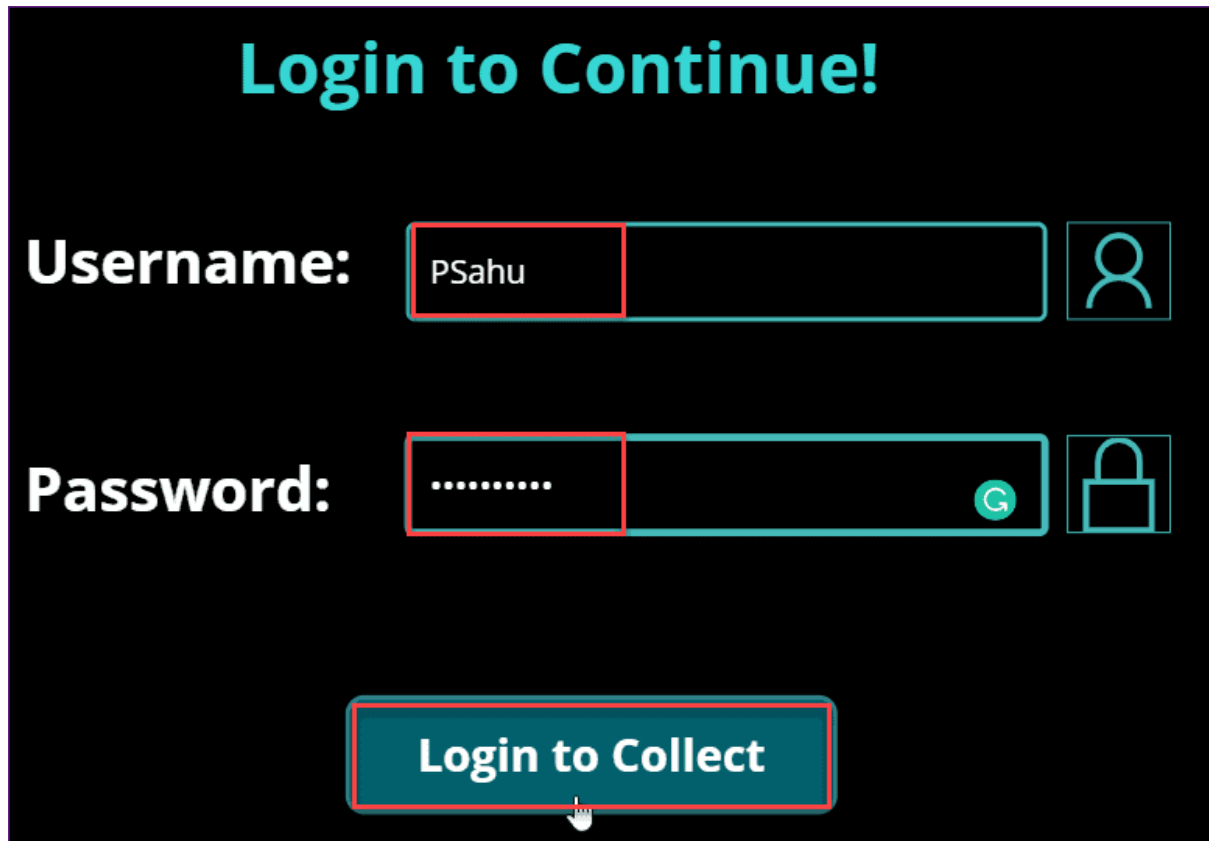
Where,

- **Collect** = Collect is a function in Powerapps that helps to store the data in Powerapps memory
- **UserLogin** = Collection name where the User login details will store
- **UserName** = It specifies the column name that will store all the user names
- **txtUsername** = Username text input control name
- **Password** = It specifies the column name that will store all the user's Password
- **txtPassword** = Password text input control name






PowerApps login formula

- Now save and preview (F5) the app. Enter **Username**, **Password**, and then hit on the **Login to Collect** button.



Login to Continue!

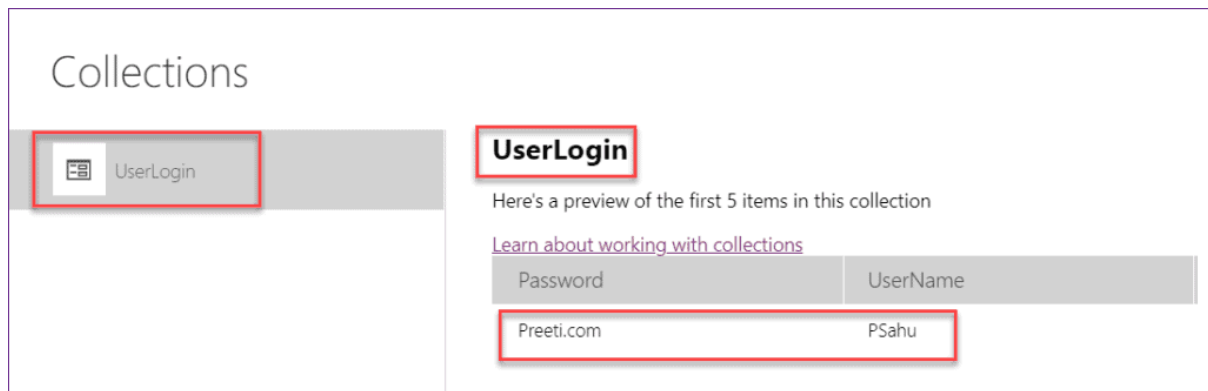
Username: 

Password:  

Login to Collect

microsoft powerapps login screen

- Go to the Powerapps Collections (**View -> Collections -> Click** on Collection name [**UserLogin**]), there you can see your Login user credentials has been stored as like the below screenshot:



Here we learned, how to **create a custom login page or screen in PowerApps**.

Also, we saw how to **create a Powerapps Login page or screen using SharePoint List** and steps to Create a Login page in PowerApps Collection.

Where,

- **txtPassword** = Password text input control name
- **"TSInfo"** = I set the Password as TSInfo. Not only TSInfo but also you can set anything here inside the " ".
- **Navigate** = Navigate is the function that helps to change from one screen to another screen.
- **'Welcome Screen'** = This is the Screen name. When the user will successfully login without having any error, then this screen will appear.
- **ScreenTransition.None** = This is the transition argument where the new screen quickly replaces the current screen.
- **Notify** = Notify function helps to display a notification message to the end-user on the Power apps screen.

- **“Invalid Password!!! Please Enter a Valid Password”** = This is the error notification message that will display when a user will enter an invalid password.
- **NotificationType.Error** = This notification argument specifies the display message is an error message.

Step-4:

- Just save and preview (F5) the app. Enter the Username (any name) and Password as **TSInfo** and hit on the **LOGIN** button. Here, the Password input field is appearing with “.....”, because I set its Mode property to Password.
- If the Password validation is correct, then at the same time the Welcome screen will appear as shown below.