Power Apps- Form Field Validations

Form validation is process where a form checks if the information provided by a user is correct or not. The form will either alert the user that they messed up and need to fix something to proceed, or the form will be validated, and the user will be able to continue with their form submission process.

In our case, we have a Power Apps form with following fields:

Name: Name of employee – Text field

Employe ID: ID of employee – Number field

Email Address: Email field

Phone Number: Text field with specific format- ###-###-####

Age: Number field with value greater than 21

Joining Date: Date field, greater than current date

Password: Password field

Here is the Form with fields:



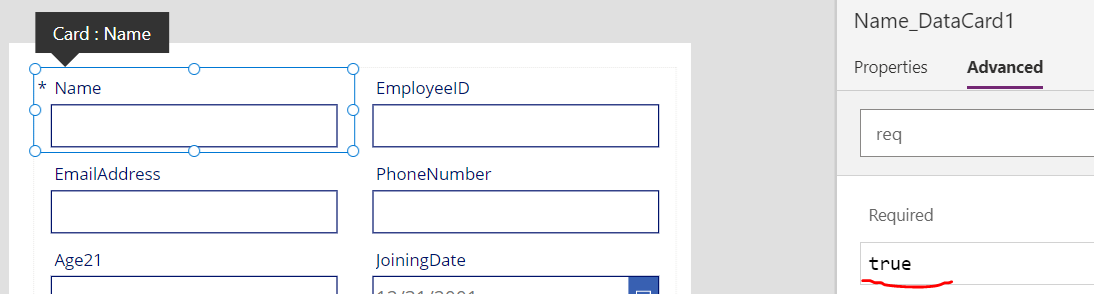
Below are the steps and approaches to validate these different type of fields in Power Apps.

## Validation for Name Field – Required field validation

Select the Name card. From the advanced properties choose to unlock to change properties



On the Required property of the card change it to be true



This is the PowerApps required field validation on submit.

For implementing validation before submit, go to Fill property of Name text field and add validation for IsBlank to show yellow background if field is empty.

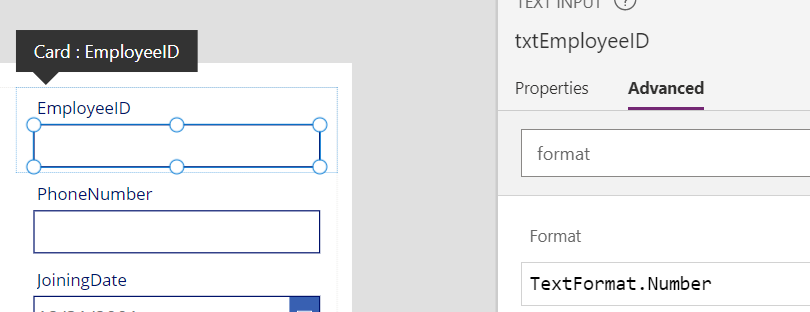
Fill: ***If(IsBlank(txtName),Yellow, RGBA(255, 255, 255, 1))***



Tooltip: ***If(!IsBlank(txtName.Text), "", "The Field can not be blank")***

## Validation for Employee ID Field – Numeric field validation

To validate Employee textbox to accept only numbers, select the textbox and make the Format property to be number.

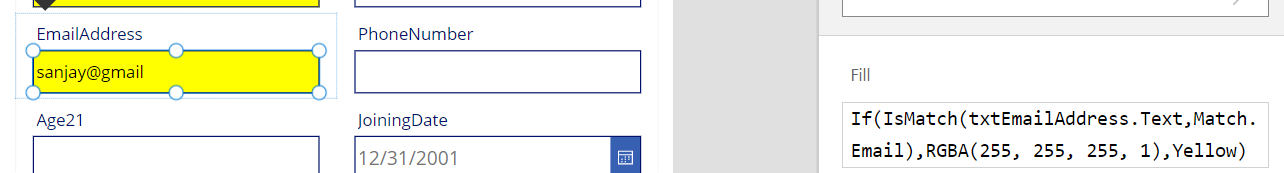


## Validation for Email Address Field – Email format field validation

An email address must be in the format test@testdomain.com. There is a predefined matching pattern already exists for email addresses that you can use.

Select the Email Address textbox. In the Fill property write this formula

Fill**: If(IsMatch(txtEmailAddress.Text,Match.Email),RGBA(255, 255, 255, 1),Yellow)**





## Validation for Phone Number Field – Phone number format validation

Select the Phone number textbox. Make the Format property to be **text** to allow digits and - characters. In the Fill property set the formula to allow only 10 digits in the format ###-###-####

**If(IsMatch(txtPhoneNumber.Text, Match.Digit&Match.Digit&Match.Digit&"-"&Match.Digit&Match.Digit&Match.Digit&"-"&Match.Digit&Match.Digit&Match.Digit&Match.Digit),RGBA(255, 255, 255, 1),Yellow)**

In Tooltip:

**If(**

**IsMatch(**

**txtPhoneNumber.Text,**

**Match.Digit&Match.Digit&Match.Digit&"-"&**

**Match.Digit&Match.Digit&Match.Digit&"-"&**

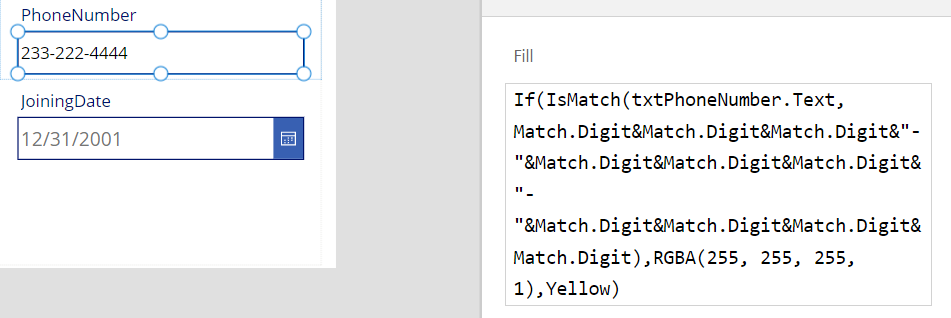
**Match.Digit&Match.Digit&Match.Digit&Match.Digit**

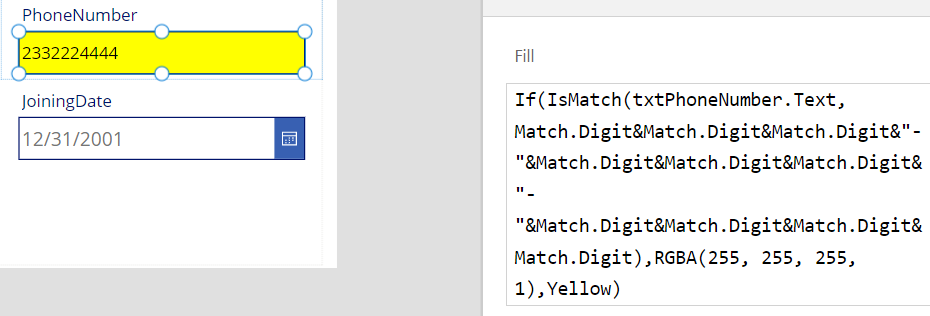
**),**

**"",**

**"The Phone number should be in the format ###-###-#### "**

**)**





## Validation for Age Field – Numeric field with conditional validation

The age field is more than 21 years old. Select the Age textbox. Make the Format property to be number to allow only digits.

Tooltip: ***If(Value(txtAge.Text) >= 21, "", "Age must be >=21")***

Fill: **If(Value(txtAge.Text) >=21,RGBA(255, 255, 255, 1),Yellow)**





## Validation for Joining Date Field – Date field validation for specific days in future

The user must choose a weekday from Monday to Friday and must be a day in the future.

**joinDateErrorMessage.Text** as well as **joinDate.Tooltip**:

**If(**

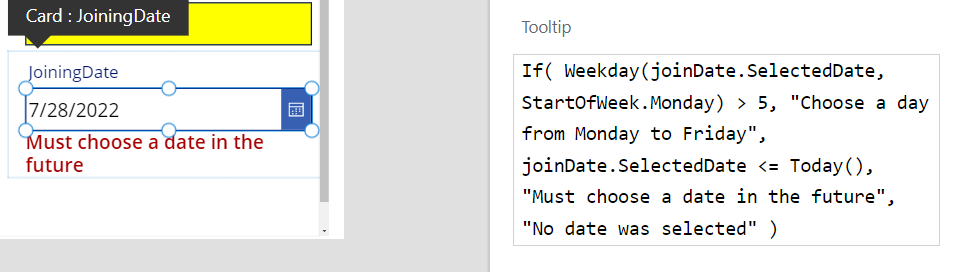
**Weekday(joinDate.SelectedDate, StartOfWeek.Monday) > 5,**

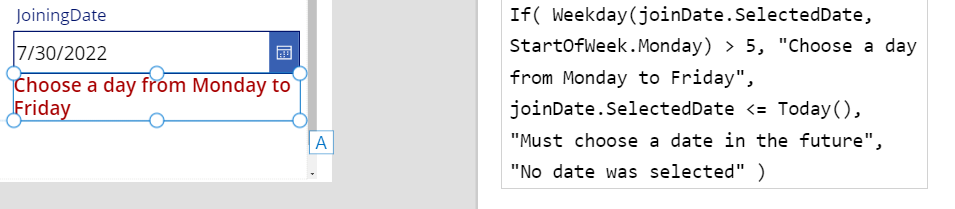
**"Choose a day from Monday to Friday",**

**joinDate.SelectedDate <= Today(),**

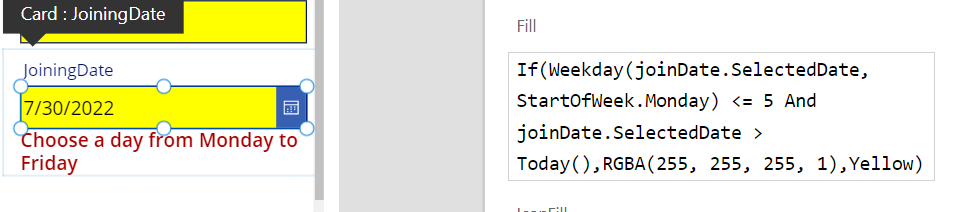
**"Must choose a date in the future"**

**)**





Fill: **If(Weekday(joinDate.SelectedDate, StartOfWeek.Monday) <= 5 And joinDate.SelectedDate > Today(),RGBA(255, 255, 255, 1),Yellow)**



## Validation for Password Field – Password format validation

Select the Password textbox. Insert a check or any Icon in the text field at set it at the right corner. In the Icon property set the formula to be:

Icon:

**If(**

**IsMatch(**

**txtPassword.Text,**

**"^(?=[^\d\_].\*?\d)\w(\w|[!@#$%]){7,20}"**

**),**

**Check,**

**Icon.Lock**

**)**

this validates a strong password which can contain eight, nine or 10 characters with the addition of at least one digit and at least one alphabetic character and no special characters.

In the color property of Icon write this formula:

**If(Self.Icon=Icon.Check, Green, Red)**

