



## Les contrôles FormPublisher

Le quoi ?

Les contrôles FormPublisher sont des fonctions mises à disposition dans FormPublisher qui, ajoutées à un champ, permettent en mode html de contrôler les [données](#) saisies par l'utilisateur.

Pourquoi ce [Document](#) ?

Il va permettre aux utilisateurs de prendre connaissance des contrôles que dispose FormPublisher ainsi que leurs intérêts d'utilisation.

Le comment ?

Ce document liste, définit et décrit tous les contrôles (check\_input) disponibles dans FormPublisher.

## 1. Définition Générale

Qu'est ce qu'un control FormPublisher ?

Un control FormPublisher est une fonction disponible dans FormPublisher qui :

- peut être ajouté à un champ de saisie
- peut être paramétrée
- permet de contrôler (valider ou non valider) les [données](#) saisies en mode [HTML](#)

FormPublisher dispose de plusieurs Contrôles repartis en plusieurs catégories.

Nous distinguons:

1. Les contrôles ou check input **numériques**
2. Les contrôles ou check input **monétaires**
3. Les contrôles ou check input **dates**
4. Les contrôles ou check input **web**
5. Les contrôles ou check input **texte**
6. Les contrôles ou check input **format**
7. Les contrôles ou check input **Belges**
8. Les contrôles ou check input **Luxembourgeois**
9. Les assistants

## 2. Contrôles numériques

Les contrôles numériques peuvent être appliqués aux **champs** de saisies de type entier (La valeur de l'attribut `DataType` est *integer*).

### 2.1. Check input number

Le control *Number* permet de contrôler que les **données** saisies sont de type entier.

[Exemple](#)

### 2.2. Check input sum

Supposons avoir dans un **formulaire** plusieurs **champs** de **données** de type entier ( `DataType = integer` ) auxquels est affecté le control *Number*.

Le control *Sum* affecté à un autre champ, calcule la somme des valeurs des champs appartenant au même groupe.

Option

Le Paramètre *Format*: ce paramètre est facultatif. S'il est renseigné, la somme est arrondie au format spécifié.

Obligation

*Groupe*: il permet d' identifier les variables concernées par le calcul de la somme

[Exemple](#)

### 2.3. Check input lessThan

Ce control permet :

1. De vérifier que la valeur saisie est un nombre valide
2. D'arrondir la valeur saisi en fonction d'un certains format (si celui-ci est spécifié)
3. De Vérifier que le nombre saisi est strictement inférieur à tous ceux du même groupe portant le control *greaterThan*

Obligation

*groupe*: permet de spécifier le groupe d'appartenance

Option

*Format*: S'il est renseigné, le nombre est arrondie au format spécifié.

[Exemple](#)

## 2.4. Check input greaterThan

Ce control permet :

1. De vérifier que la valeur saisie est un nombre valide
2. D'arrondir la valeur saisie en fonction d'un certain format (si celui-ci est spécifié)
3. De vérifier que le nombre saisi est strictement supérieur à tous ceux du même groupe portant le control *lessThan*

Obligation

*groupe*: permet de spécifier le groupe d'appartenance

Option

*Format*: S'il est renseigné, le nombre est arrondi au format spécifié.

[Exemple](#)

## 2.5. Check input lessOrEqualThan

Ce control permet :

1. De vérifier que la valeur saisie est un nombre valide
2. D'arrondir la valeur saisie en fonction d'un certain format (si celui-ci est spécifié)
3. De vérifier que le nombre saisi est inférieur ou égal à tous ceux du même groupe portant le control *greaterOrEqualThan*

Obligation

*groupe*: permet de spécifier le groupe d'appartenance

Option

*Format*: S'il est renseigné, le nombre est arrondi au format spécifié.

[Exemple](#)

## 2.6. Check input greaterOrEqualThan

Ce control permet :

1. De vérifier que la valeur saisie est un nombre valide
2. D'arrondir la valeur saisie en fonction d'un certains format (si celui-ci est spécifié)
3. De Vérifier que le nombre saisi est supérieur ou égal à tous ceux du même groupe portant le control *lessOrEqualThan*

Obligation

*groupe*: permet de spécifier le groupe d'appartenance

Option

*Format*: S'il est renseigné, le nombre est arrondie au format spécifié.

[Exemple](#)

## 2.7. Check input percentage

Le control *percentage* permet de vérifier que la [donnée](#) saisie est nombre compris entre 0 et 100.

[Exemple](#)

## 2.8. Check input sumPercentage

Ce control permet :

1. De faire la somme des pourcentages portant le contrôle *percentage* et ayant le même groupe
2. De Vérifier que la somme ne dépasse pas 100.

Obligation

*groupe*: permet de spécifier le groupe d'appartenance

[Exemple](#)

### 3. Contrôles monétaires

Les contrôles monétaire sont applicables aux [champs](#) de saisies de type entier (La valeur de l'attribut DataType est *integer*).

#### 3.1. Check input currency

Ce control permet :

1. formater un nombre monétaire.

Option

*Format*: paramètre facultatif. S'il est renseigné, il indique le format de la devise.

Le caractère " \* " indique la position de la devise.

Exemple:

\*£ : indique que la devise est après le montant

\$\* : indique la devise est avant le montant

*Groupe*: paramètre facultatif. il permet d'identifier le groupe d'appartenance

[Exemple](#)

#### 3.2. Check input sumCurrency

Ce control permet :

1. calcule et formate la somme des montants d'un groupe.

Option

*Format*: il indique le format de la devise.

*Groupe*: il permet d' identifier les variables concernées par le calcul de la somme

[Exemple](#)

## 4. Contrôles date

Les contrôles dates peuvent être appliqués aux [champs](#) de saisies de type chaîne de caractères (La valeur de l'attribut `DataType` est *string*).

### 4.1. Check input date

Ce control permet :

1. De définir le format de la date
2. De contrôler la validité de la date saisie

#### Obligation

*Format*: il indique le format de la date à contrôler.

- deux digits pour définir le jour
- deux digits pour définir le mois
- quatre digits pour définir l'année

Exemple: dd/mm/yyyy ---> 25/01/2011

#### Option

*showAssistant*: permet d'afficher ou non l'assistant calendrier. La valeur par défaut est: *true*

[Exemple](#)

### 4.2. Check input startDate

Ce control permet :

1. De définir le format de la date
2. De contrôler la validité de la date saisie
3. De vérifier que la date saisie est antérieur à toute date appartenant au même groupe et ayant le contrôle *date*

#### Obligation

*Format*: il indique le format de la date à contrôler.

- deux digits pour définir le jour
- deux digits pour définir le mois
- quatre digits pour définir l'année

Exemple: dd/mm/yyyy ---> 25/01/2011

*groupe*: permet identifier le groupe d'appartenance

#### Option

*showAssistant*: permet d'afficher ou non l'assistant calendrier. La valeur par défaut est: *true*

[Exemple](#)

### 4.3. Check input endDate

Ce control permet :

1. De définir le format de la date
2. De contrôler la validité de la date saisie
3. De vérifier que la date saisie est postérieur à toute date appartenant au même groupe et ayant le contrôle *date*

#### Obligation

*Format*: il indique le format de la date à contrôler.

- deux digits pour définir le jour
- deux digits pour définir le mois
- quatre digits pour définir l'année

Exemple: dd/mm/yyyy ---> 25/01/2011

*groupe*: permet identifier le groupe d'appartenance

#### Option

*showAssistant*: permet d'afficher ou non l'assistant calendrier. La valeur par défaut est: *true*

[Exemple](#)

### 4.4. Check input heure

Ce control permet :

1. De définir le format de la date
2. De contrôler la validité de l'heure saisie (vérifie que l'heure saisie est comprise entre 0 et 23).

#### Obligation

*Format*: il indique le format de l'heure à contrôler.

- deux digits pour définir le l'heure
- deux digits pour définir les minutes
- deux digits pour définir les secondes

Exemple: hh:mm:ss ---> 12:03:00

[Exemple](#)

## 5. Contrôles internet

Les contrôles internet peuvent être appliqués aux **champs** de saisies de type chaîne de caractères (La valeur de l'attribut `DataType` est *string*).

### 5.1. Check input eMail

Ce control permet :

1. De vérifier la validité de l'adresse email saisie.

[Exemple](#)

### 5.2. Check input url

Ce control permet :

1. De vérifier la validité d'une adresse de site web (url)

[Exemple](#)

## 6. Contrôles textes

Les contrôles texte peuvent être appliqués aux [champs](#) de saisies de type chaîne de caractères (La valeur de l'attribut `DataType` est *string*).

### 6.1. Check input rangeChar

Ce control permet :

1. De vérifier que le texte saisi contient le nombre minimum ou le nombre maximum de [caractère](#) requis

Obligation

*min*: permet de définir le nombre de caractère minimum requis.

*max*: permet de définir le nombre de caractère maximum requis.

[Exemple](#)

### 6.2. Check input rangeCharWithCounter

Ce control permet :

1. De vérifier que le texte saisi contient le nombre minimum ou le nombre maximum de [caractère](#) requis
2. D'afficher le nombre de caractères déjà saisis

Obligation

*min*: permet de définir le nombre de caractère minimum requis.

*max*: permet de définir le nombre de caractère maximum requis.

[Exemple](#)

### 6.3. Check input regExp

Ce control permet :

1. De valider un champ par rapport à une expression régulière.

Obligation

*regExp*: définit l'expression régulière

[Exemple](#)

## 7. Contrôle format

Ce control permet :

1. De formater en fonction du format défini, la chaîne de caractères saisie

Obligation

*Format:* il indique le format de saisi.

- Le # représente les digits
- Le x représente les lettres
- Le \* représente n'importe quels autres caractères

Exemple: xx\*#####xx##### ----> CE/1250/AB2011

[Exemple](#)

## 8. Contrôles lists

Les contrôles list peuvent être appliqués aux listes

### 8.1. Check input uniqueOnly

Ce control permet :

1. D'afin d'empêcher la saisie de doublons dans les listes.

Obligation

*key*: permet de définir le(s) champ(s) sur le(s)quel(s) la vérification sera faite. ( Il peut y avoir plusieurs paramètres *key* pour vérifier l'unicité de la liste)

[Exemple](#)

## 9. Contrôles officiels

### 9.1. Contrôles belges

Nous définissons ici les contrôles de l'administration belge disponibles dans FormPublisher.

#### 9.1.1. Check input BEBCE

Ce control permet :

1. De contrôler la validité des numéros BCE belges .

[Exemple](#)

#### 9.1.2. Check input BEONSS

Ce control permet :

1. De contrôler la validité des numéros d'immatriculation à l'ONSS belge .

[Exemple](#)

#### 9.1.3. Check input BENISS

Ce control permet :

1. De contrôler la validité des numéros NISS belges .

[Exemple](#)

#### 9.1.4. Check input BETVA

Ce control permet :

1. De formater et de contrôler la validité des numéro TVA belges .

[Exemple](#)

#### 9.1.5. Check input BEBankNumber

Ce control permet :

1. De formater et de contrôler la validité des numéros bancaires belges .

[Exemple](#)

#### 9.1.6. Check input BEPhone

Ce control permet :

1. De contrôler la validité des numéros de téléphone belges .

[Exemple](#)

## 9.2. Contrôles luxembourgeois

Nous définissons ici les contrôles de l'administration luxembourgeoise disponibles dans FormPublisher.

### 9.2.1. Check input LUMatriculePM

Ce control permet :

1. De contrôler la validité du numéro matricule [personne](#) morale luxembourgeois .

[Exemple](#)

### 9.2.2. Check input LUMatriculePP

Ce control permet :

1. De contrôler la validité du numéro matricule [personne](#) physique luxembourgeois .

[Exemple](#)

## 9.3. Autres Contrôles

### 9.3.1. Check input BIC

Ce control permet :

1. De contrôler la validité des numéros BIC (Bank Identifier Code) .

[Exemple](#)

### 9.3.2. Check input IBAN

Ce control permet :

1. De contrôler la validité des numéros IBAN (International Bank Account Number) .

[Exemple](#)

## 10. Contrôle général

### 10.1. Logical

Il permet la mise en erreur d'un champ en fonction d'une expression booléenne passée en paramètre.

Le champ portant le control est mis en erreur si l'expression n'est pas vérifiée.

Obligation

*Expression*: il permet de saisir l'expression booléenne de test.

[Exemple](#)

## 11. Contrôle upload

### 11.1. Check input LimitFileSize

Permet de **mettre en erreur** un champ de type "**uploadBox**" si la pièce chargée dépasse la taille définie par le paramètre " *maxSizeAllowed*" exprimée en octets.

Obligation

*maxSizeAllowed*: Permet de définir la taille maximale de la pièce à charger .

[Exemple](#)

## 12. Les assistants

### 12.1. Assistant helperCPV

helperCPV est l'assistant de saisie du code postal

Obligation

*id*: indique le genre d'assistant.

Exemple: cpvWol

*groupe*: définit le groupe d'appartenance. Ceci permet de relier les [champs](#) portant respectivement les assistants helperCPV et helperVOC

[Exemple](#)

### 12.2. Assistant helperVOC

helperVOC est l'assistant de saisie des libellés

Obligation

*id*: indique le genre d'assistant.

Exemple: cpvWol

*groupe*: définit le groupe d'appartenance. Ceci permet de relier les [champs](#) portant respectivement les contrôles helperCPV et helperVOC

[Exemple](#)

### 12.3. Assistant helperNUTS

helperNUTS est l'assistant de saisie des codes NUTS

[Exemple](#)

---