# Instructions to Setup Custom Fields in The Juniper IITBombayX

**for the Juniper OpenedX release**

# These instructions are created for the Sysad team to add custom fields in the registration form in the Juniper release of IITBombayX portal. Four custom fields are added in the registration form.

1. **state**: mandatory drop down list of the state in India,
2. **city**: mandatory text field,
3. **pincode**: mandatory text field which allowed only 6-digit numerical value and
4. **aadharid**: optional text field which allowed 12-digits numerical value.

# Steps to follow

1. Make sure the Internet connection is proper using the following command.

***lynx internet.iitb.ac.in***

1. Clone the application from the git repository using the following command:

***git clone {git repository} {example AppDirectory}***

Use the following command:

***git clone http://<username>@gitlab.cse.iitb.ac.in/IITBombayXCustomRegForm/IITBomabayXCustomRegForm.git***

1. Go to the application directory

***cd IITBomabayXCustomRegForm/***

1. Checkout the git branch “*IITBombayX”* using the following command

***git checkout IITBomabayX-Juniper***

1. Change the ownership of the application directory “*IITBomabayXCustomRegForm*” to the “*edxapp*” using the following command

***sudo chown -R edxapp:edxapp ../IITBomabayXCustomRegForm***

1. Open the “**lms.yml**” file located in the **“/edx/etc/”** directory (you may have to create it if it doesn't exist.). Please note all the keys in the lms.yml files should be in the alphabetic order.

**sudo vi /edx/etc/lms.yml**

* 1. Add *"custom\_reg\_form"* in the "*ADDL\_INSTALLED\_APPS*" sequences. After adding, it should look like this

***ADDL\_INSTALLED\_APPS:***

***- custom\_reg\_form***

* 1. Set a "*REGISTRATION\_EXTENSION\_FORM*" to "*custom\_reg\_form.forms.ExtraInfoForm*". After setting the above parameter, it should look like this.

***REGISTRATION\_EXTENSION\_FORM: custom\_reg\_form.forms.ExtraInfoForm***

* 1. Make sure the value of “***city****”* is “***hidden***” and “***state*” must be removed** ( should not be present) in the "*REGISTRATION\_EXTRA\_FIELDS*" mapping in the “**lms.yml**” file, It should look like this

***REGISTRATION\_EXTRA\_FIELDS:***

***city: hidden***

***confirm\_email: hidden***

***country: required***

***gender: optional***

***goals: optional***

***honor\_code: required***

***level\_of\_education: optional***

***mailing\_address: hidden***

***terms\_of\_service: hidden***

***year\_of\_birth: optional***

* 1. Save and exit from the file.

1. In the same directory “**/edx/etc/**”, open the “**studio.yml**” file (you may have to create it if it doesn't exist.). Please note all the keys in the syudio.yml files should be in the alphabetic order.

**sudo vi /edx/etc/studio.yml**

and set the “*ENABLE\_COMBINED\_LOGIN\_REGISTRATION*” feature flag to **“true**” in the “FEATURES” mapping. After setting the flag it should look like this.

**FEATURES:**

**AUTH\_USE\_OPENID\_PROVIDER: true**

**AUTOMATIC\_AUTH\_FOR\_TESTING: false**

**CUSTOM\_COURSES\_EDX: true**

**ENABLE\_COMBINED\_LOGIN\_REGISTRATION: true**

Save and exit from the file.

1. **Install the application:**
   1. Make sure the Internet connection is proper using the following command.

***lynx internet.iitb.ac.in***

* 1. Go to the directory “*/home/edx*”

Make sure the current working directory is “*/home/edx*” using the following command.

***Pwd***

If it is not “*/home/edx*”, change it to this directory using the following command.

***cd***

* 1. Change the user “edxapp” by executing the following command on the shell:

***sudo -H -u edxapp bash***

* 1. Activate edx platform virtual environment “*edxapp\_env” using the following command*

***source /edx/app/edxapp/edxapp\_env***

* 1. Now, Install this application using following command:

***pip install -e* IITBomabayXCustomRegForm*/***

* 1. Exit from the “*edxapp*” user shell using the following command:

***exit***

1. **Run migrations:**
   1. Make sure the Internet connection is proper using the following command.

***lynx internet.iitb.ac.in***

* 1. Go to the directory “*/home/edx*”

Make sure the current working directory is “*/home/edx*” using the following command.

***Pwd***

If it is not “*/home/edx*”, change it to this directory using the following command.

***cd***

* 1. Change the user *“edxapp*” by executing the following command on the shell:

***sudo -H -u edxapp bash***

* 1. Activate edx platform virtual environment “*edxapp\_env”* using the following command

***source /edx/app/edxapp/edxapp\_env***

* 1. Go to the “*edx-platform*” directory using following command:

***cd /edx/app/edxapp/edx-platform/***

* 1. Run Migrations using following command:

***./manage.py lms migrate custom\_reg\_form --settings=production***

* 1. Exit from the “*edxapp*” user shell using the following command:

***exit***

1. **Start/restart** the LMS service using the following command.

***sudo /edx/bin/supervisorctl restart lms***

Once the setting up of custom fields is complete, verification is required. We need to verify the database and registration process

## Check the MySQL Database

Connect to mysql database using the following command

**mysql -u root -p**

After logging in to mysql, go to **edxapp** database using following command

***Use edxapp;***

Run the following queries in *edxapp* database:

1. Check if the table "*custom\_reg\_form\_extrainfo*" is created with the following fields: **“id”, “state”, “city”, “pincode”, “aadharid”** and **“user\_id”**.

***desc custom\_reg\_form\_extrainfo;***

1. Check if a row is added with the “**custom\_reg\_form**” app in "**django\_migrations**" table

**select \* from django\_migrations where app = 'custom\_reg\_form';**

Results will be displayed in a row for the app “*custom\_reg\_form*” with file name “***0001\_initial***” in the name column.

1. Check if a row is added with a **'custom\_reg\_form'** app in **"django\_content\_type"** table.

**select \* from django\_content\_type where app\_label ='custom\_reg\_form';**

Result will be displayed with the columns as :**id, app\_label** and **model**

1. Check if *Add, Change, Delete and View permissions* (4 rows) are added for **'django\_content\_type'** of **"custom\_reg\_form"** app in **"auth\_permission"** table.

**select \* from auth\_permission where content\_type\_id in (<list\_of\_ids\_result\_of\_above\_step3>);**

**<list\_of\_ids\_result\_of\_above\_step3>** is the data displayed in column name “ *Id* ” from the **"django\_content\_type"** table after running the query mentioned in step 3.

Eg.: *select \* from auth\_permission where content\_type\_id in (304, 305);*

Result will be displayed with the columns as : ***id, name, content\_type\_id*** *and* ***codename***

“*codename*” column will display **Add\_<model>, Change\_<model>, Delete\_<model>** and **view\_\_<model> permission** for each *content\_type\_id*( same as  *id* from step 3). For each **Id**, 4 rows will be there displaying *Add, Change, Delete and View*.

## B) Check the registration form:

1. Go to the url of IITBombayX and register a user by filling the registration form along with **“state”, “city”, “pincode”** and **“aadharid”** fields.
2. Check if the data of the registered user is saved in a table in the MySQL database.
   1. Go to ***edxapp*** database by using the command

**use edxapp;**

* 1. Get the **id** of the registered user from “*auth\_user*” table by running the following query

**select max(id) from auth\_user;**

By running this query, we get the ***id*** of the latest row entered. That will be the registered *user’s id*. This **Id** will be used for checking the required registration details.

* 1. For the above max(id), check **"auth\_userprofile"** table for user profile details (year\_of\_birth, gender etc) using the queries written below:

**select \* from auth\_userprofile where user\_id = <max(id)>;**

The entries should be present in the **auth\_userprofile** table**.**

* 1. For the above *max(id)*, check **"custom\_reg\_form\_extrainfo"** table for **state name, city name, pincode and aadharid** entries using the queries written below:

**select \* from custom\_reg\_form\_extrainfo where user\_id = <max(id)>;**

The entries should be present in the **custom\_reg\_form\_extrainfo** table**.**