Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра информационных систем и технологий

**Лабораторная работа № 4**

«Инфраструктура Docker»

Выполнил:

Студент: Станчик М.А.

ФИТ 3 курса 4 группы

Преподаватель: Сазонова Д.В.

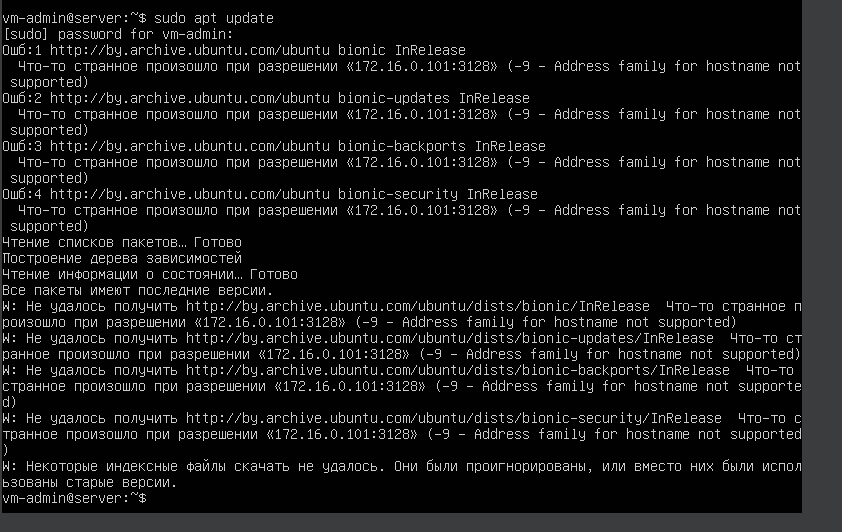
Минск 2025

**Цель:** освоить навыки развертывания контейнера из образа и его настройки в среде docker, передачи файлов между операционной системой и контейнером docker, запуска интерактивных приложений внутри контейнера и управления контейнером.

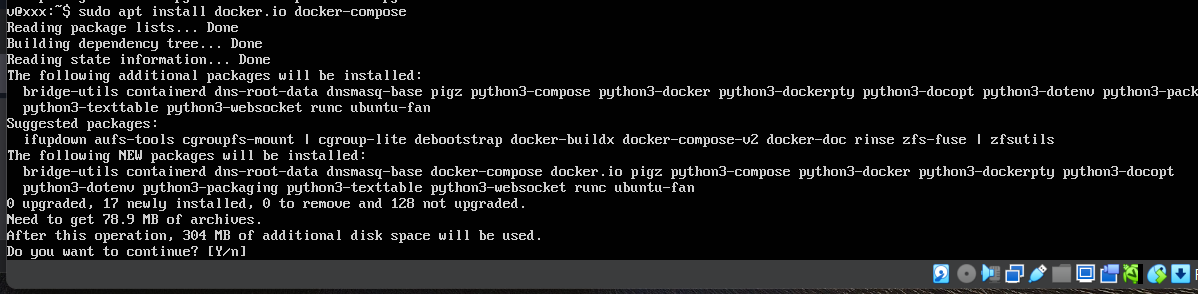
**Задачи:**

1. **на виртуальной машине через утилиту apt установить клиентский пакет docker и серверный пакет dockerd;**

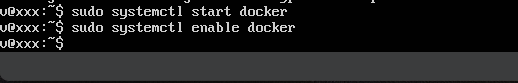
|  |
| --- |
| sudo apt update - Обновите пакеты на виртуальной машине |

****

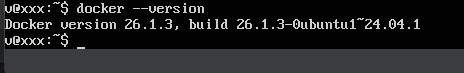
|  |
| --- |
| sudo apt install docker.io docker-compose - Установите необходимые пакеты для работы с Docker |

****

|  |
| --- |
| sudo systemctl start docker - Запустите службу Docker  sudo systemctl enable docker - Включите службу Docker |

****

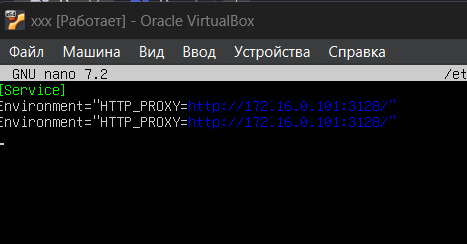
|  |
| --- |
| docker –version - Проверьте, что Docker установлен и работает |

****

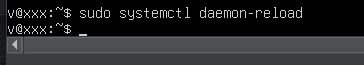
1. **. при использовании университетской сети сконфигурировать dockerd на использование прокси-сервера 172.16.0.101:3128;**

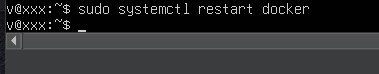
|  |
| --- |
| sudo mkdir -p /etc/systemd/system/docker.service.d  sudo nano /etc/systemd/system/docker.service.d/http-proxy.conf - Создайте или отредактируйте файл конфигурации Docker |

****

****

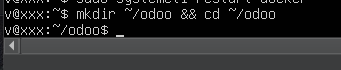
|  |
| --- |
| sudo systemctl daemon-reload  sudo systemctl restart docker - Перезагрузите конфигурацию Docker и перезапустите службу |

****

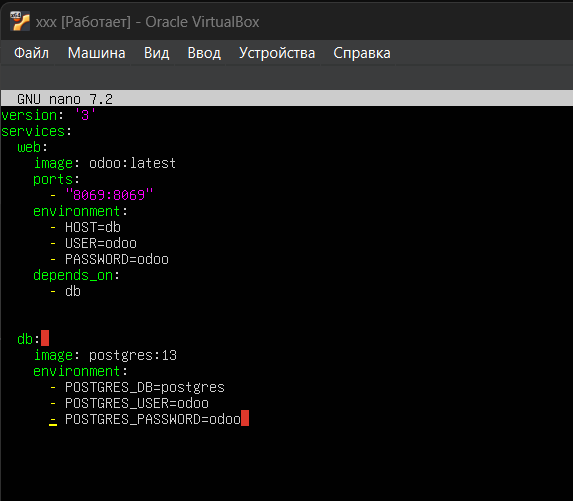
****

1. **выполнить установку ERP-системы odoo с помощью docker и инструкций на сайте hub.docker.com, согласиться на внесение в базу данных демонстрационных данных;**

|  |
| --- |
| mkdir ~/odoo && cd ~/odoo - Создайте директорию для проекта Odoo |

****

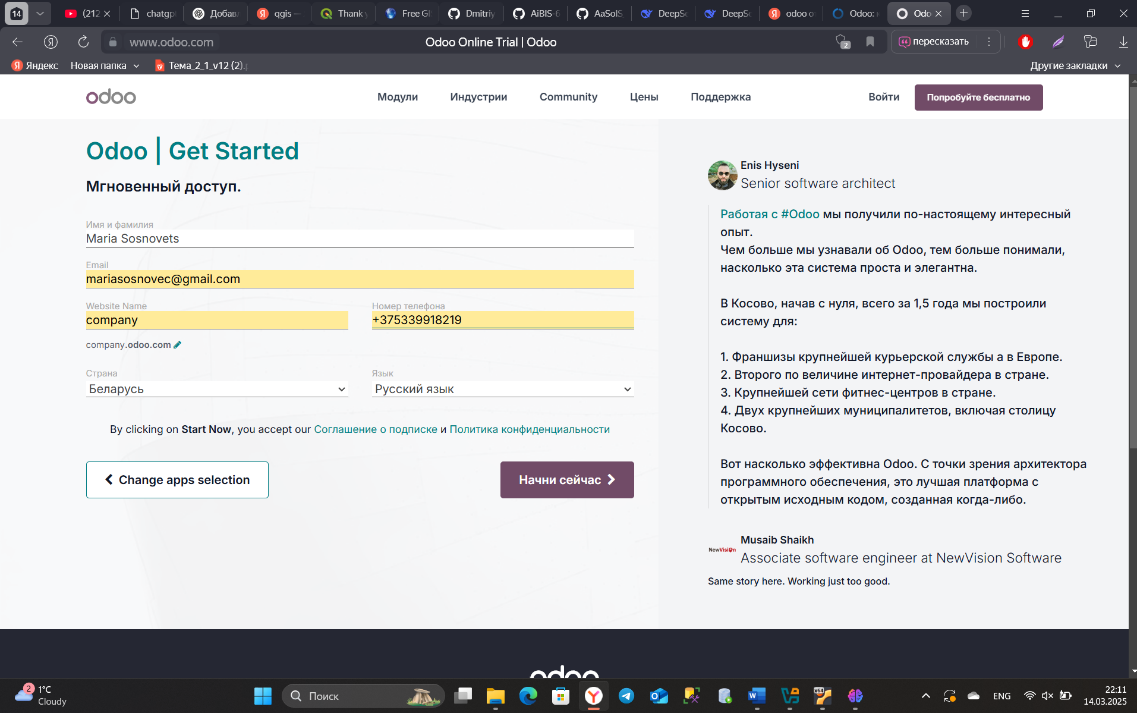
|  |
| --- |
| Создайте файл docker-compose.yml |

****

|  |
| --- |
| docker-compose up -d - Запустите контейнеры |

****

1. **сконфигурировать ERP-систему odoo на использование SMTP-сервера postfix, входящего в состав пакета iRedMail, для отправки e-mail;**
2. Войдите в админ-панель Odoo.
3. Перейдите в Настройки > Технические > Электронная почта > SMTP-серверы.
4. Добавьте новый SMTP-сервер с параметрами:
   * SMTP-сервер: postfix (или IP-адрес вашего сервера Postfix)
   * Порт: 25
   * Логин и Пароль: если требуется аутентификация, укажите данные от почтового ящика.



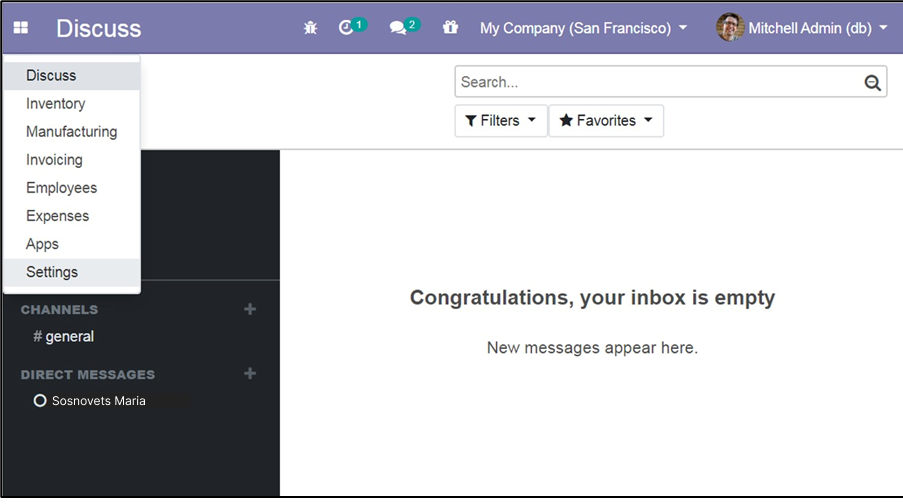
1. **сконфигурировать SMTP-сервер postfix на прием писем из docker-контейнера с ERP-системой odoo;**

Настройте Postfix на сервере, чтобы он принимал соединения от Docker-контейнера. Отредактируйте файл /etc/postfix/main.cf

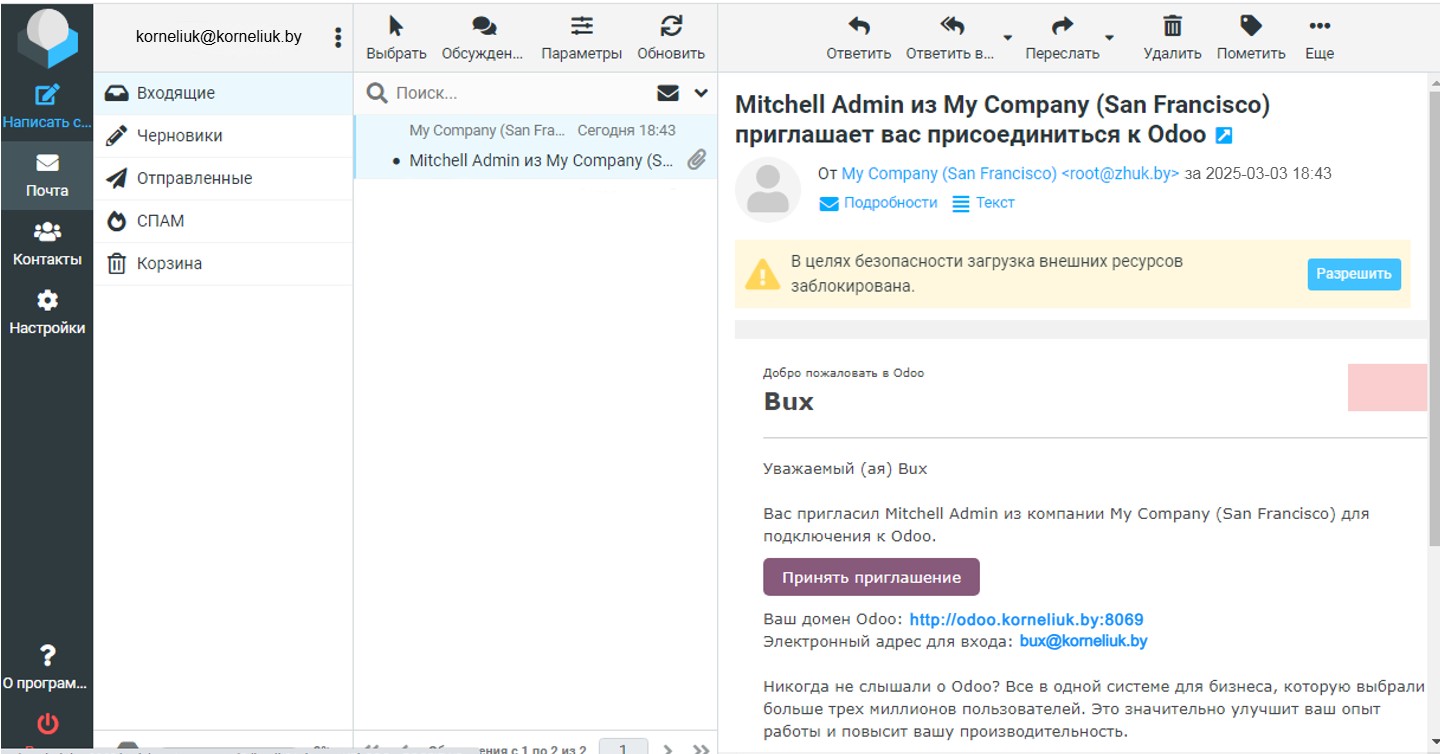
|  |
| --- |
| mynetworks = 127.0.0.0/8 [::1]/128 172.17.0.0/16 |

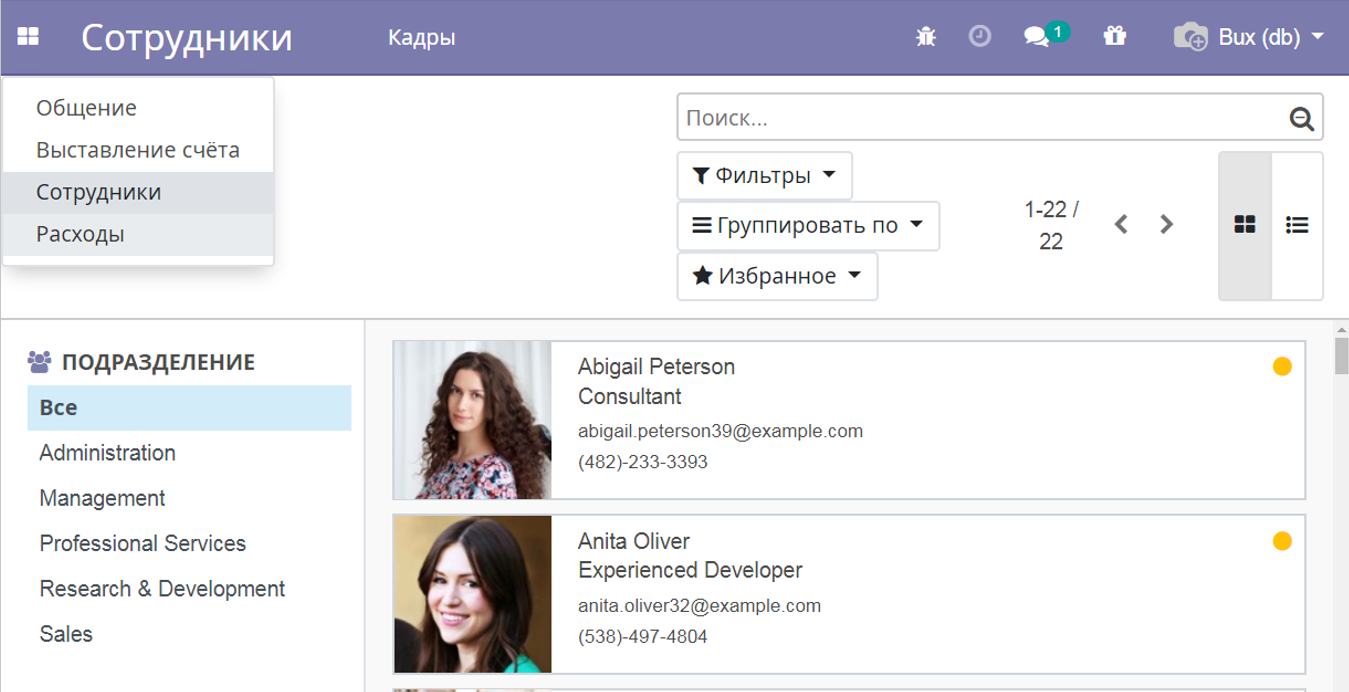
|  |
| --- |
| sudo systemctl restart postfix - Перезапустите Postfix |

1. **создать почтовый ящик на почтовом сервере с помощью панели iRedAdmin;**

****

1. **создать пользователя в ERP-системе odoo, назначить ему права на использование только бухгалтерских компонентов, выслать подтверждение на созданный почтовый ящик;**
2. Войдите в Odoo как администратор.
3. Перейдите в Пользователи > Создать.
4. Заполните данные пользователя, назначьте права только на бухгалтерские компоненты.
5. Укажите почтовый адрес, созданный в iRedAdmin, и отправьте приглашение.
6. **выполнить регистрацию пользователя в ERP-системе odoo, перейдя по ссылке в полученном на почтовый ящик письме и заполнив регистрационную форму.**

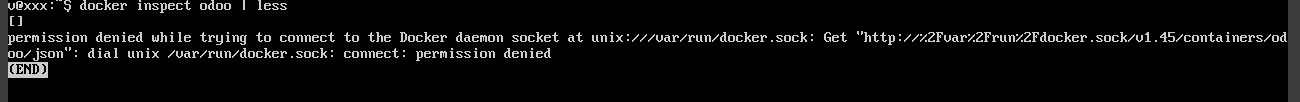




1. **определите пути в ОС Ubuntu и внутри контейнера odoo к папке, используемой odoo для размещения дополнительных модулей, с помощью команды**

docker inspect odoo | less

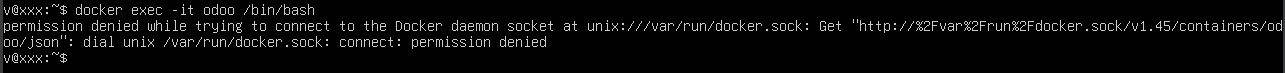
|  |
| --- |
| docker inspect odoo | less |



1. **запустите командную строку bash внутри контейнера командой**

docker exec -it odoo /bin/bash

|  |
| --- |
| docker exec -it odoo /bin/bash |



1. **создайте заготовку модуля в папке для дополнительных модулей командой**

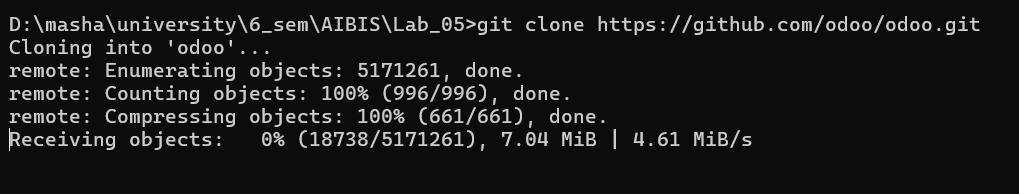
odoo scaffold test\_module /mnt/extra-addons/

1. **выйдите из командной строки внутри контейнера командой**

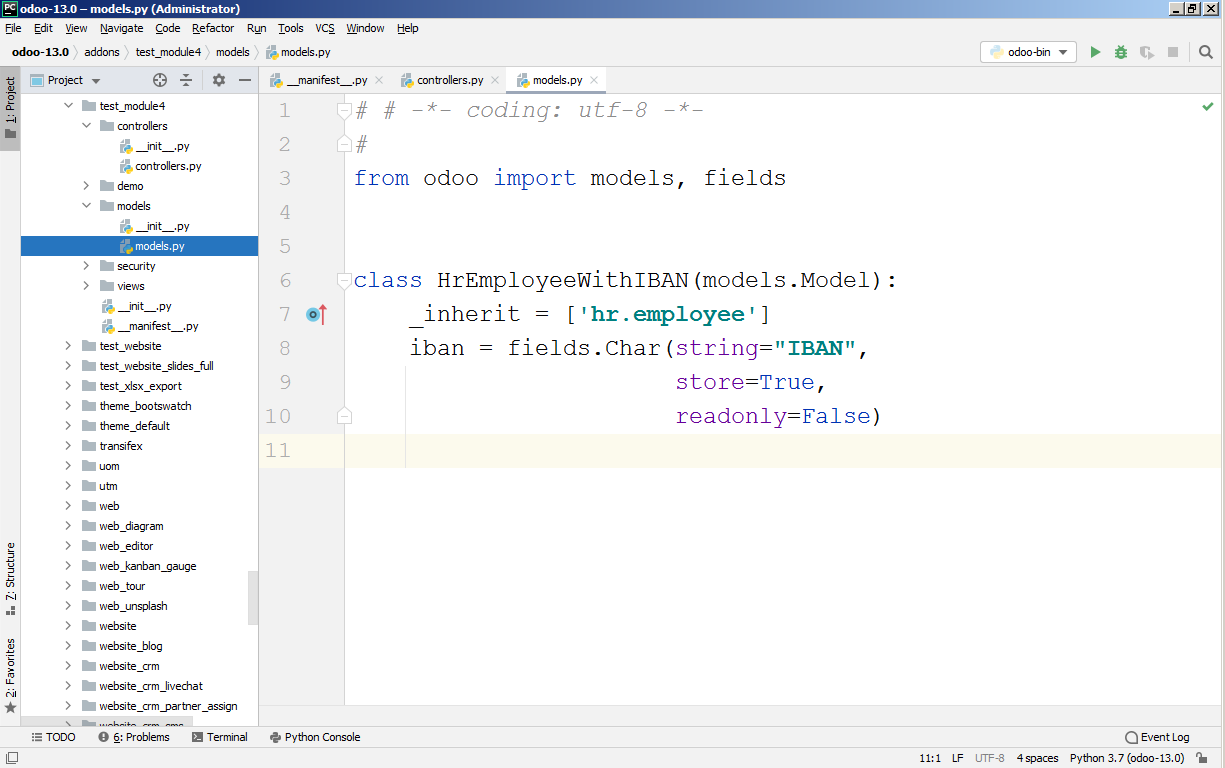
exit

1. **на физической машине для удобства редактирования исходного кода модуля скачайте исходный код odoo c github:**

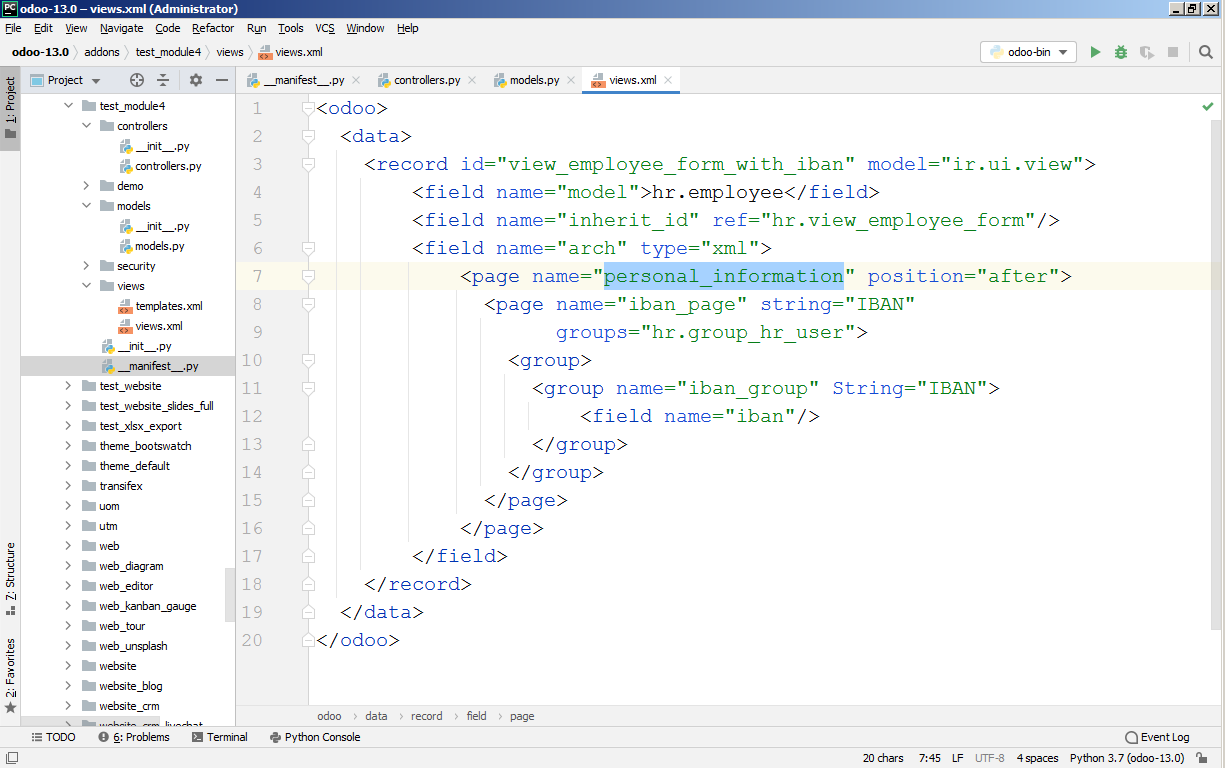
<https://github.com/odoo/odoo>



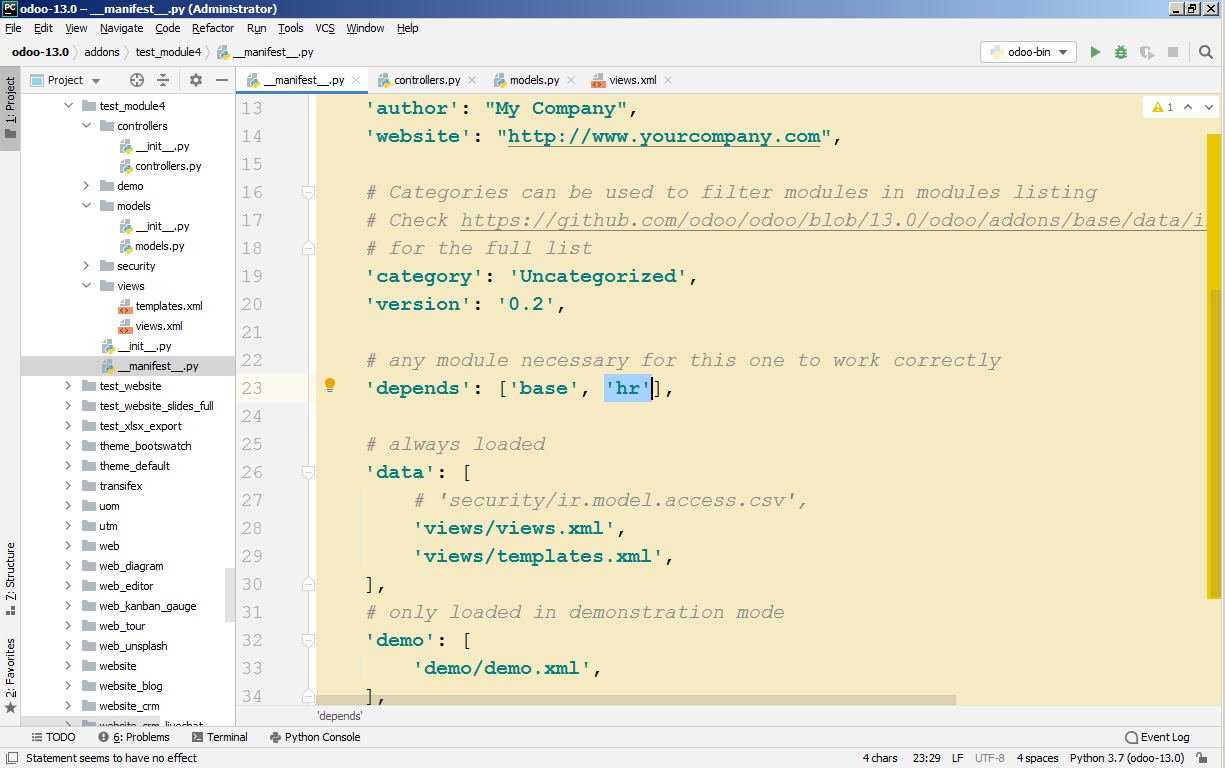
1. **распакоуйте скачанный исходный код в отдельную папку;**
2. **скопируйте по FTP с виртуальной машины созданную ранее заготовку модуля (папку)** test\_module **на физическую машину в папку** /addons **внутри папки с исходным кодом odoo; не перепутайте с папкой** /odoo/addons**;**
3. **для редактирования исходного кода модуля на языке Python целесообразно скачать и установить среду разработки PyCharm Community Edition, при установке поставьте галочку «Add “Open Folder as Project”»;**
4. **нажмите правой кнопкой мыши по папке с исходным кодом odoo и выберите пункт «Open Folder as PyCharm Community Edition Project»;**
5. **на панели Project найдите заготовку модуля (папку);**
6. **в первую очередь необходимо обеспечить хранение номеров счетов работников с помощью модели, поэтому отредактируйте файл** /models/models.py **внутри папки с заготовкой модуля следующим образом для добавления поля** iban **к существующей модели** hr.employee**:**

****

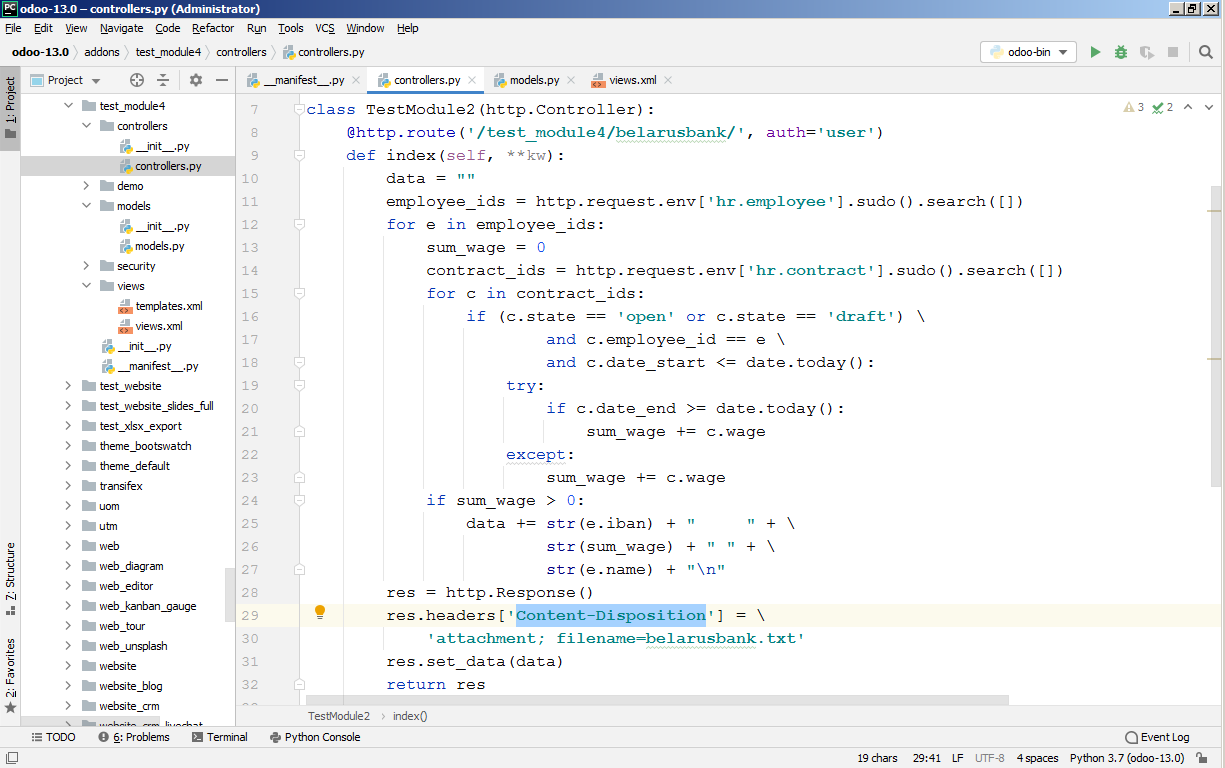
1. **для обеспечения возможности редактирования номера счета на веб-страницах odoo добавьте вкладку с полем для ввода/вывода номера счета в существующее представление** hr.view\_employee\_form**, например, после вкладки** personal\_information**, отредактировав файл** /views/views.xml **следующим образом:**

****

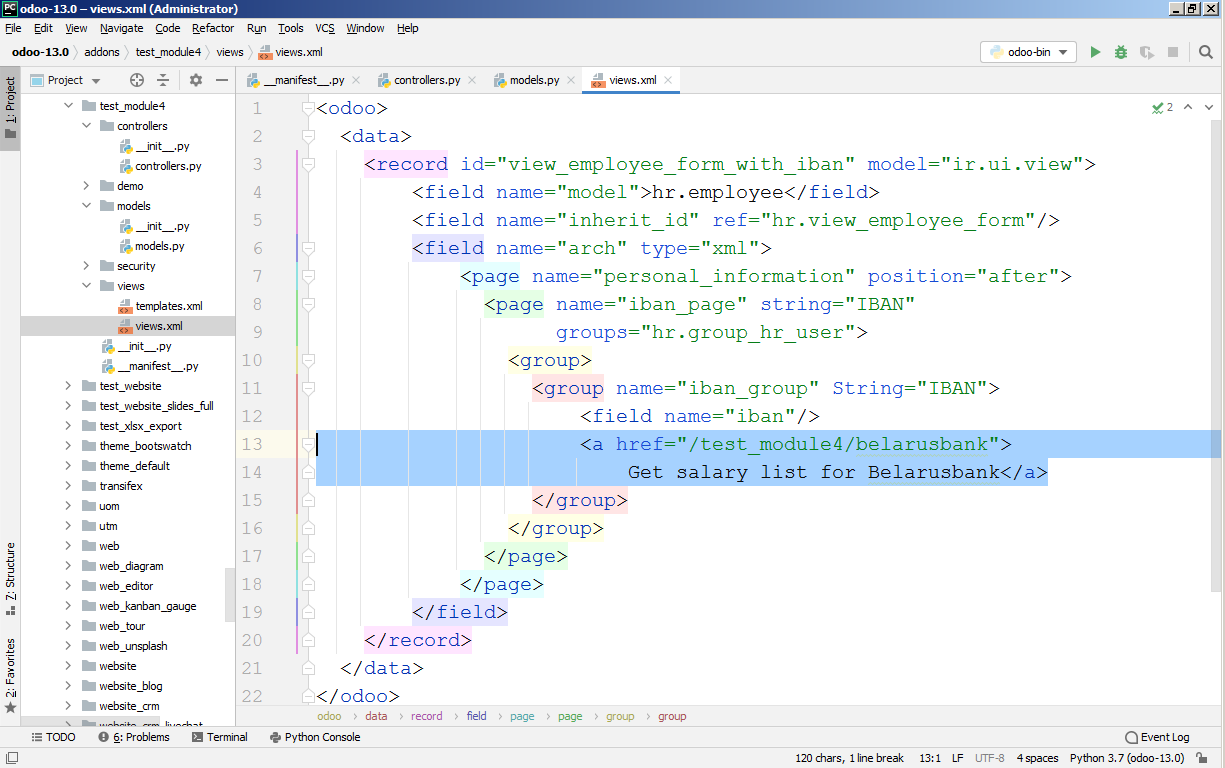
1. **поскольку наше расширяющее представление привязывается к модели из модуля** hr**, укажите данный модуль в списке зависимостей нашего модуля в файле** \_\_manifest\_\_.py**:**

****

1. **для импорта в клиент-банк требуется генерировать простой текстовый файл, поэтому требуется использовать не представление, а контроллер; создайте в файле** /controllers/controllers.py **контроллер с привязанным к определенному URL методом** index**, перебирающим всех работников, суммирующим для каждого работника оплату по всем действующим на текущую дату контрактам, формирующим файл из трех колонок, разделенных пробелами (IBAN, сумма выплат по контрактам, ФИО работника) и добавляющим перед файлом заголовок**Content-Disposition **для загрузки этого файла, а не открытия в браузере:**

****

1. **добавьте в созданное ранее представление в файле** /views/views.xml **ссылку на метод созданного контроллера:**

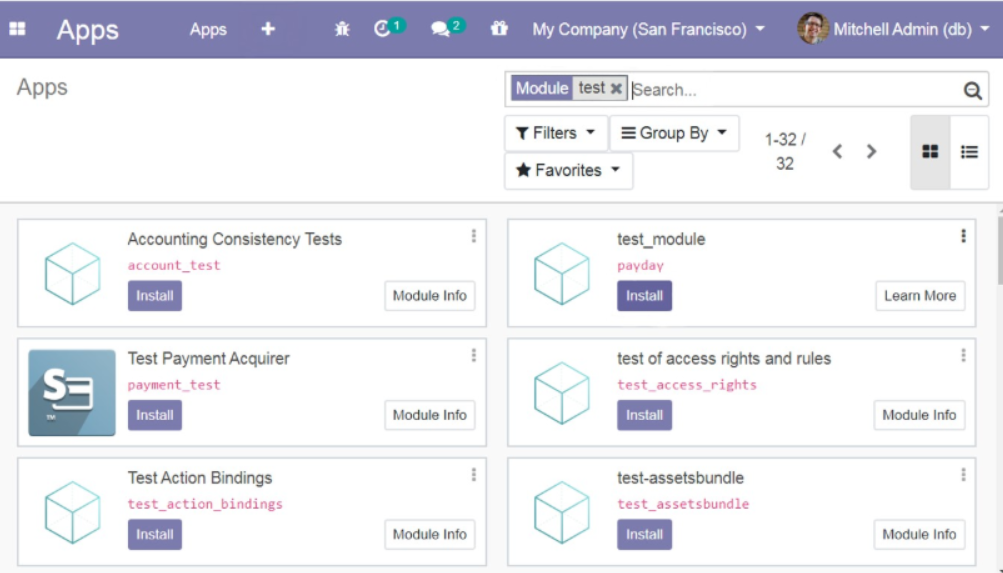
****

1. **сохраните все отредактированные файлы;**
2. **передайте отредактированный модуль обратно на виртуальную машину, в папку для дополнительных модулей odoo; при отсутствии прав на запись выдайте права на папку командой** chmod **всем пользователям;**
3. **для загрузки информации о новом модуле в odoo перезапустите контейнер командами**

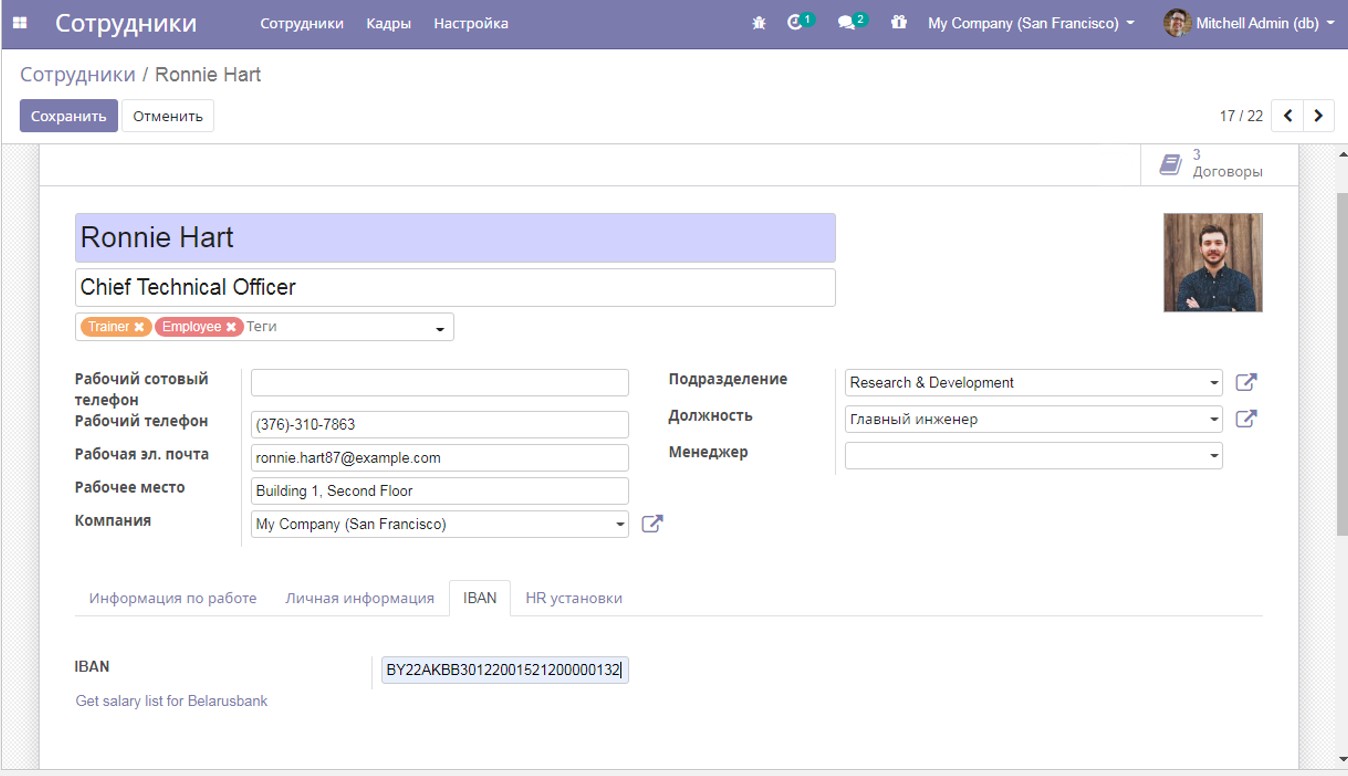
docker stop odoo

docker start odoo

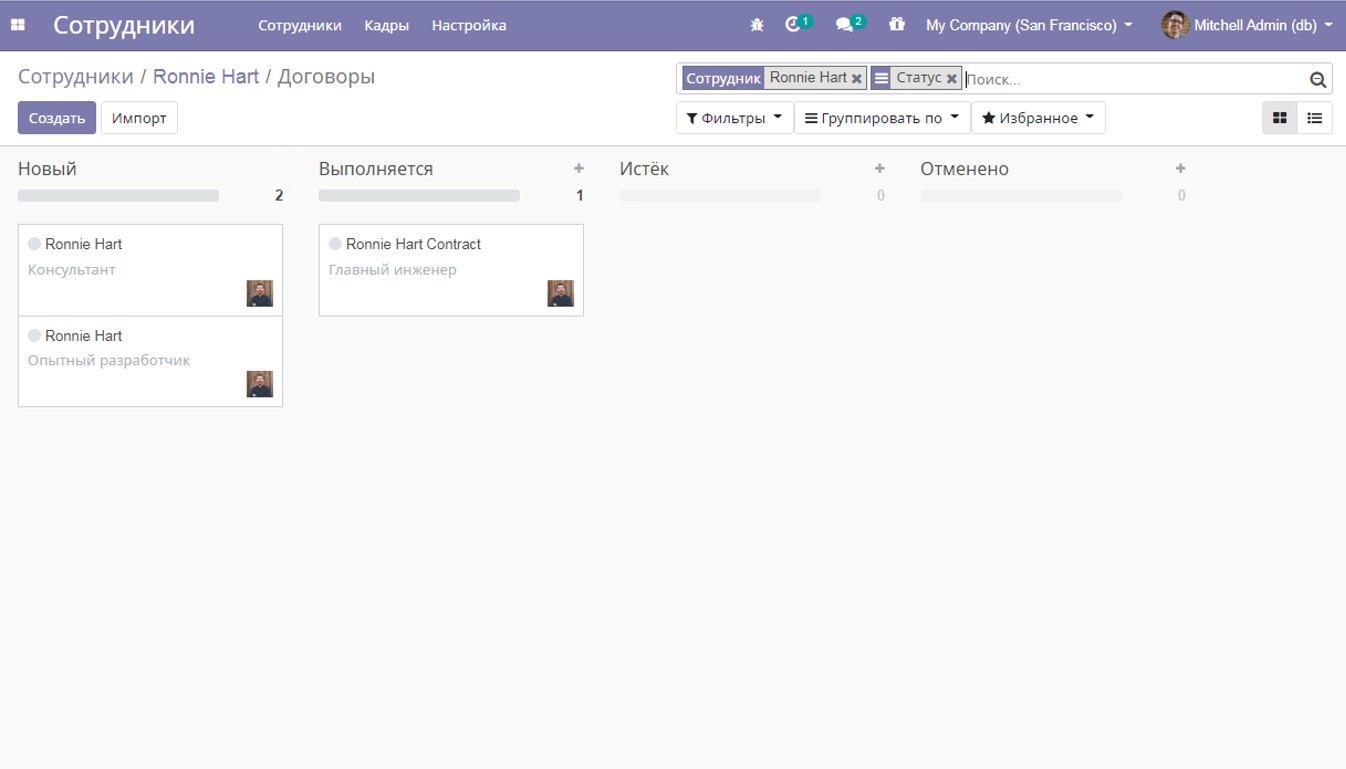
1. **для установки собственного модуля включите режим разработчика в настройках odoo;**
2. **установите созданный модуль через пункт главного меню «Приложения», найти свой модуль можно по названию с помощью панели поиска;**



1. **найдите сотрудников с действующими на текущую дату контрактами и внесите им IBAN;**



1. **найдите хотя бы одного сотрудника с несколькими действующими на текущую дату контрактами или добавьте новый контракт сотруднику с одним действующим контрактом;**



1. **нажмите на ссылку «Get salary list for Belarusbank» и убедитесь в корректности формирования списка на выплату зарплат: сотрудник с несколькими действующими контрактами не должен повторяться, для него должна быть указана сумма выплат по всем действующим контрактам.**

