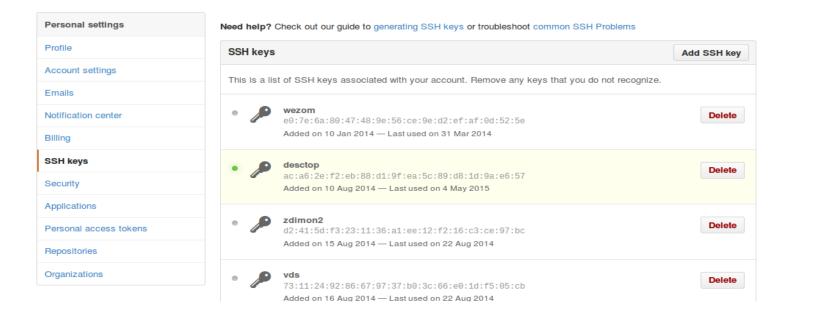
Урок 2

GIT

init clone add commit pull push branch tag reset status log

генерация ключика ssh-keygen -t rsa cat ~/.ssh/id rsa



Основы языка.

```
Строка это как и все в питоне - объект.
Обьект класса string.
Раз объект, то должны быть методы.
Методы подстановки значения
s = 'hello %s' % 'dima'
d = 'hello %s your number is %d' % ('dima', '4')
s = \text{'bla } \{0\} \text{ bla } \{1\}'.\text{format}(['1', '2'])
'%.2f' % amount
Запись в файл
f = open('text.txt', 'w')
f.write(s)
f.close()
```

Перечисляемые типы

Кортеж (1,2,3)

Список [1,2,3]

Словарь { '1' : 'one', '2' : 'two' }

Срезы

```
list = [1,2,3,4,5,6,7,8,9]
v = list[2:5] # [4,5,6,7]
v = list[:2] # [1,2]
v = list[3:] # [4,5,6,7]
```

Установщик пакетов рір

pip install packet_name requirements.txt - зависимости перечисление пакетов

Django==1.6.5 django-modeltranslation==0.7.3 Установка пакетов pip install -r requirements

Работаем с сервером

```
Найти подстроку grep -Ri "text-to-find-here" /
```

- i stands for upper/lower case (optional in your case).
- R stands for recursive.

```
netstat -pl | grep 8000
kill
chmod +x filename.py
```

Основы bash shell

cd . ~

mkdir

Любой bash-скрипт должен начинаться со строки: #!/bin/bash

\$1 - первый параметр

\$0 - имя скрипта

есho - вывод

exit 0 - выход

Зарезервированные переменные

\$НОМЕ - домашний каталог пользователя

\$HOSTNAME - ваш hostname

\$HOSTTYPE - архитектура машины.

\$РАТН - путь поиска программ

Оптимизация

```
#!/bin/bash
comment=$1
dt=`date '+%d/%m/%Y_%H:%M:%S'`
git add --all
if [ -n "$comment" ]
  then
    git commit -m "commit at $dt"
  else
    git commit -m "$comment"
fi
git push
```

Django команды

- ./django-admin.py startproject o
 ./manage.py startapp main
- ./manage.py syncdb
- ./manage.py runserver 8888
- http://ip:8888/admin

Структура проекта

settings.py - настройки urls.py - роутинг app/models.py - модели app/views.py - вьюхи

Первое приложение с моделью

```
./manage.py startapp blog
blog/models.py
#coding: utf-8
from django.db import models
class BlogCategory(models.Model):
      name = models.CharField(max_length=250, verbose_name=u'3аголовок', blank=False)
      description = models.TextField(verbose name=u'Описание')
      class Meta:
            verbose_name = u'Категория'
            verbose name plural = u'Категории'
```

./manage.py syncdb

Подключение приложения в админку.

```
### blog/admin.py
#coding: utf-8
from django.contrib import admin
from blog.models import BlogCategory

class BlogCategoryAdmin(admin.ModelAdmin):
    pass
```

admin.site.register(BlogCategory,BlogCategoryAdmin)

Переопределяем главную

```
### blog/view.py
#coding: utf-8
from django.template import RequestContext
from django.shortcuts import render to response
def home(request):
   return render_to_response('home.html', {},context_instance=RequestContext(request))
### blog/templates/home.html
<h1>Hello my friend!</h1>
### myapp/url.py
url(r'\^$', 'blog.views.home', name='home'),
```

Наследование шаблонов

```
## templates/base.html
{% load staticfiles %}
<html>
<head>
<link rel="stylesheet" href="{% static 'css/bootstrap.css' %}" type="text/css" media="screen" />
<title>Блог</title>
</head>
<body> {% block base content %} {% endblock %} </body>
</html>
### blog/templates/home.html
{% extends "base.html" %}
{% block base content %}<h1>Hello my friend!</h1>{% endblock %}
```