**TERMINAL**

python --version 🡪 s tem preverimo verzijo pyhtona na računalniku

pip –version 🡪 s tem preverimo pip verzijo v pythonu

pip install Flask 🡪 S tem namestimo flask v pyhton

requirements.txt – tukaj naredimo seznam paketov, ki jih naložimo na strežnik

notri vpišemo Flask (to je ime knjižnice) \*\*\* opomba, nekatera imena knjižnic se piše z veliko nekatera pa z malo

lahko pa v terminal vpišemo pip install -r requirements.txt 🡪 tako naložimo več knjižnic, ki jih mi zahtevamo

r – sporočimo, da gre za datoteko z zahtevki

od sedaj naprej, ko dajemo stvari na serverje, moramo se držat standardov – zato vpišemo v **main.py** ali **app.py**

from flask import Flask  #to je komanda za uvod flaska

#flask deluje po principu objektno orientiranega programiranja

# controler

@app.route("/")

def index():

    return "Hello"

if \_\_name\_\_=="\_\_main\_\_":

    app.run()

To je neka osnovna koda, ki je potrebna za postavitev neke spletne aplikacije – ali si jo zapomnimo, ali pa shranimo to vzorčno kodo

* uvozimo knjižnico
* kreiramo nov flask objekt
* naša vstopna stran je na statični strani vedno index.html, pri spletnih aplikacijah
* ko želi nekdo na našo stran se zažene python datoteka
* prikaz uporabniku se pokaže CONTROLER = HANDLER
* poševnica pomeni izhodišče – naprej gre program na prvi izhodišni controler in vidi, da naša funkcija samo natisne »Hello« v naš brskalnik
* Ta controler - = handler usmerja na našo spletno stran
* tale pyhton datoteka so možgani naše spletne strani
* stran ima več controlerjev oz. handlerjev – za gumbe, kontakte, obrazce, ipd.
* zelo je pomembno, kako so strukturirane naše mape - templates recimo vsebuje notri vse naše html datoteke

**Kaj smo do sedaj naredili?**

* naredili smo VIEW – to je html in Controller
* html ne more direkt komunicirati z bazo podatkov (class, objects, database) – zato imamo controller v pythonu, ki komunicira med VIEW in MODEL



če imamo spletno stran: domen.si/galerija 🡪 ta galerija je podstran, torej potrebujemo sedaj še en handler oz. controller, ki bo lahko to podstran odprl

@app.route("/galerija")

def galerija():

    return render\_template("galerija.html")

sedaj pa moramo omogočit uporabniku linke spletne strani iz ene na drugo.

če hočemo vnesti še CSS datoteke, moramo vnesti STATIC mapo, notri damo tudi slike (to dvoje se na spletni strani ne spreminja!!!)