

Foreigner project

김갑동 + 정병수 + 민희원





Agenda Style

01 설계도

전체 적인 Django system

02 개요

프로젝트의 의도

03 DB 설명 (스키마)

04 기술 구현

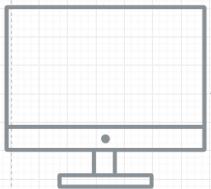
각 part 의 기술 구현 설명

05 소감

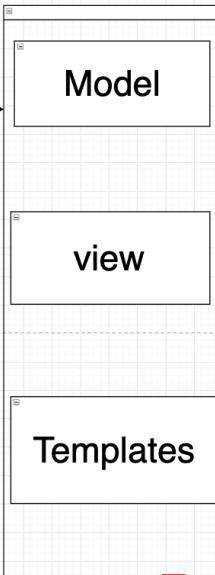
김갑동, 정병수, 민희원

설계도

Client



1.HTML 화면은 server에서 MVT과정을
끝낸 Data들이 보여지는 page이다.



server



- Orgchart
- 1.DB에 들어간 스키마를 coding 한다.
 - 2.제약 조건, insert, update, delete, select 같은 조건들은 만들어 준다.

- 1.Model에서 만들어진 데이터가 DB에 저장이 가져온다.
- 2.Data를 templates에서 사용할 수 있도록 function을 만들어 준다.(processin data를 한다.)

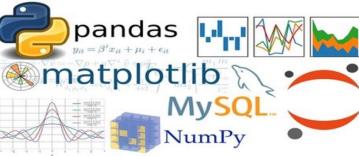
- 1.view에서 만들어진 function을 templates에서 input해준다.
- 2.HTML틀에서 원하는 곳에다가 Data를 insert해 준다.

DataBase

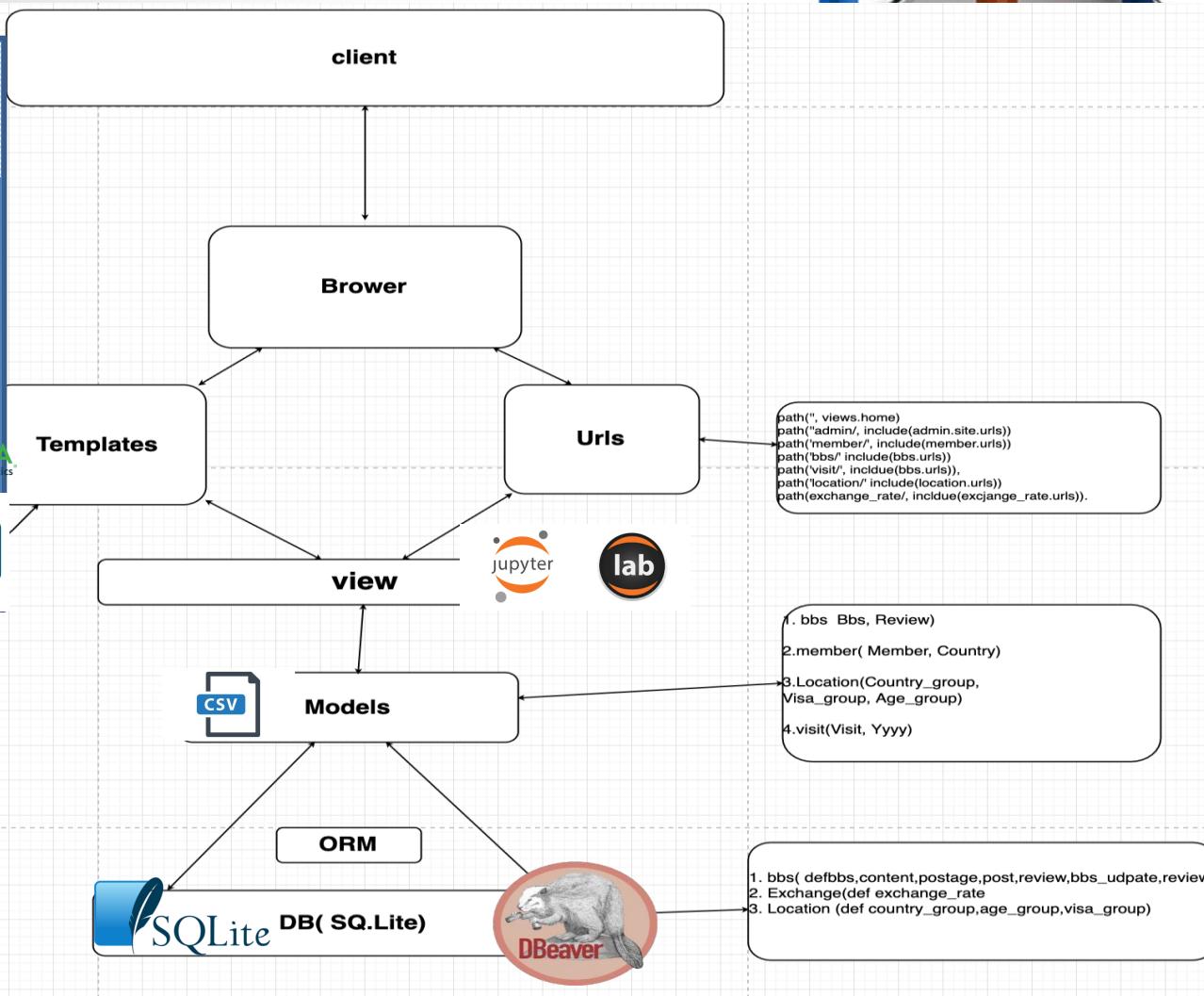


SQ Lite

- 1.server에서 insert된 data가 저장된지 확인한다.
2. DB에서 스키마를 설정할 수 도 있다 먼저 data를 (제약 조건 PK, FK, Not null, auto_increment)



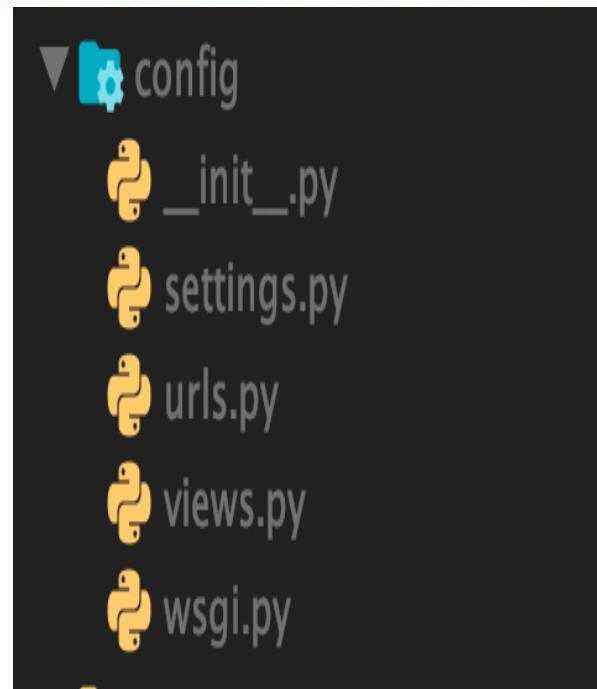
1.home.html (board,exchangerate,login,sign up)
2.base.html
3.Location(country_group content,
visa_group,age_group)
4.Member(join,login,member_info)



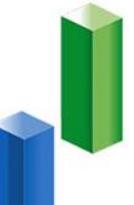


```
3
4
▼ bbs
  ▶ migrations
  ▼ templates
    ▶ bbs
      5 bbs.html
      5 content.html
      5 postPage.html
      5 update_page.html
    2 __init__.py
    2 admin.py
    2 apps.py
    2 models.py
    2 tests.py
    2 urls.py
    2 views.py
```

```
3
4
▼ member
  ▶ migrations
  ▶ static
  ▼ templates
    ▶ member
      5 join.html
      5 login.html
      5 member_info.html
    2 admin.py
    2 apps.py
    2 models.py
    2 tests.py
    2 urls.py
    2 views.py
```



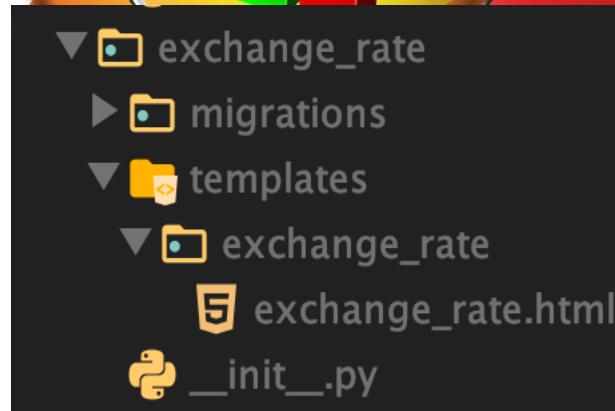
```
▼ config
  2 __init__.py
  2 settings.py
  2 urls.py
  2 views.py
  2 wsgi.py
```



```
location
migrations
static
templates
    age_group
        age_group.html
    country_group
        country_group.html
    visa_group
        visa_group.html
admin.py
apps.py
models.py
tests.py
urls.py
views.py
```



```
templates
    base.html
    home.html
venv
visit
    migrations
    static
    templates
        visit
            predict.html
            visit.html
            admin.py
            apps.py
            models.py
            tests.py
            urls.py
            views.py
```



```
exchange_rate
migrations
templates
    exchange_rate
        exchange_rate.html
        __init__.py
        admin.py
        apps.py
        models.py
        tests.py
        urls.py
        views.py
```



	단계	11/11	11/12	11/13	11/14	11/15
기획	프로젝트기획					
설계	데이터베이스설계					
코딩	화면구현					
	코딩					
구현	디버깅					

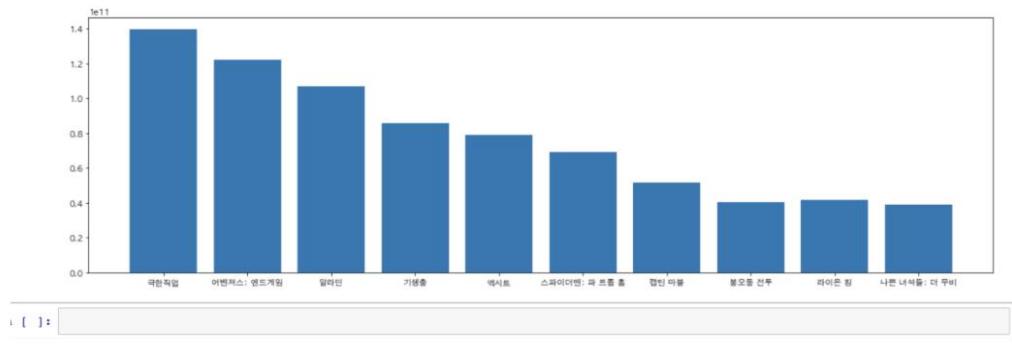
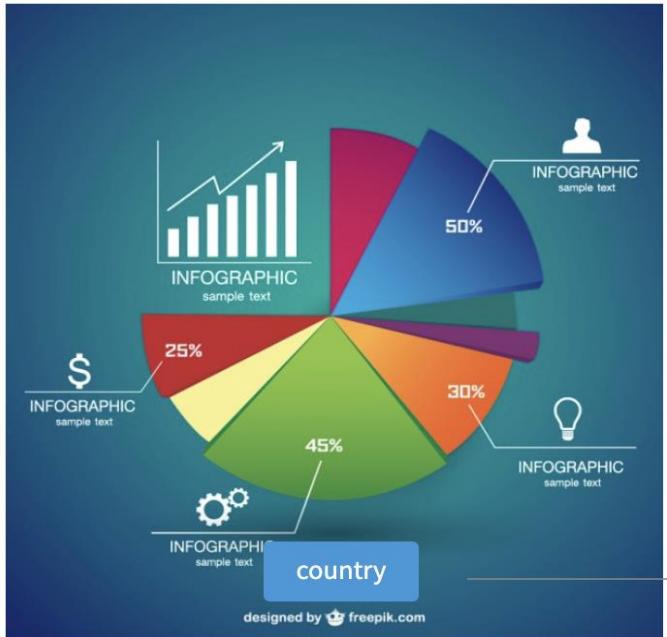
	단계	11/18	11/19	11/20	11/21	11/22
기획	프로젝트기획					
설계	데이터베이스설계					
코딩	화면구현					
	코딩					
구현	디버깅					

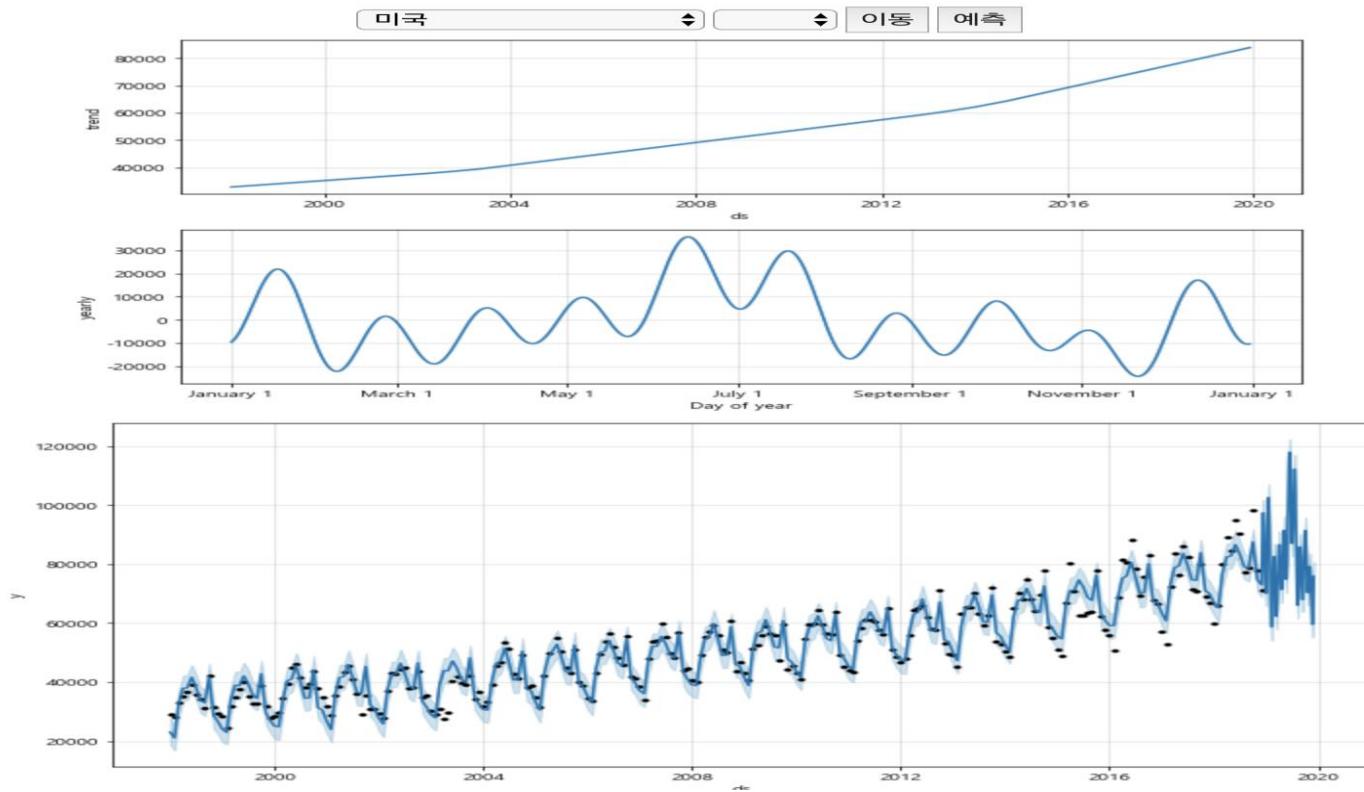
01. Visualization



A map of Seoul's districts (gu) shown as a network of white regions separated by grey lines. Some districts are highlighted with orange outlines. A thick blue line, representing a river, flows through the city. The districts labeled are: community, sign-up, edit, log-out; Gangseo-gu, Eun-pyeong-gu, Seo-daemun-gu, Dobong-gu, Nowon-gu, Gangbuk-gu, Seongbuk-gu, Jung-nang-gu, Gwangjin-gu, Gangdong-gu, Yang-cheong-deungpo-gu, Yeongdeungpo-gu, Guro-gu, Dongjak-gu, Geum-cheon-Gwanak-gu, Yongsan-gu, Jung-gu, Jongno-gu, Seong-dong-gu, Songpa-gu, and Gangnam-gu.

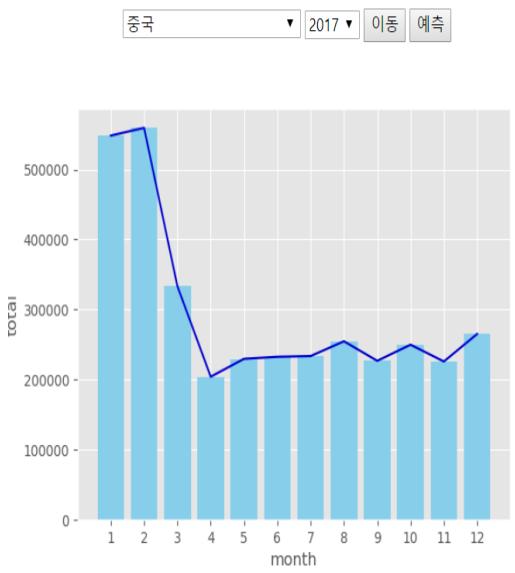






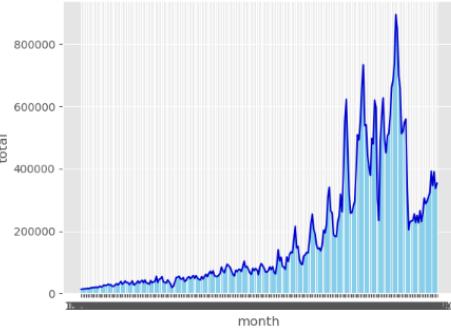


visit page



visit page

한국 2017 01 이동 예측



month	total
0 1	548289
1 2	559306
2 3	333482
3 4	204343
4 5	229851
5 6	232737
6 7	233781
7 8	254849
8 9	226997
9 10	249971
10 11	226050
11 12	265308

month	total
0 19981	13348
1 19982	13562
2 19983	15128
3 19984	13880
4 19985	16095
5 19986	15751
6 19987	16021
7 19988	18027
8 19989	17951
9 199810	18920
10 199811	19226
11 199812	19959
12 19991	18417
13 19992	23177
14 19993	20518
15 19994	21995
16 19995	26691
17 19996	25051
18 19997	26077
19 19998	29074
20 19999	27139
21 199910	27594
22 199911	22070
23 199912	23121
24 20001	27629
25 20002	30657
26 20003	27118
27 20004	32057
28 20005	38249
29 20006	29713
30 20007	34187



Welcome to Korea

[Home](#)

[board](#)

[exchange rate report](#)

[sign up](#)

[login](#)

login page

ID

PW

[로그인](#)



join page

ID

PW

PW Confirm

NcikName

Gender

Country

Email



나라를 클릭시 게시판으로 이동

□ 총 10건의 게시물

제 목 ▼

검색

번호	제 목	작성일
10	게시판 제목이 표출됩니다.	2111-11-11
09	게시판 제목이 표출됩니다.	2111-11-11
08	게시판 제목이 표출됩니다.	2111-11-11
07	게시판 제목이 표출됩니다.	2111-11-11
06	게시판 제목이 표출됩니다.	2111-11-11
05	게시판 제목이 표출됩니다.	2111-11-11
04	게시판 제목이 표출됩니다.	2111-11-11
03	게시판 제목이 표출됩니다.	2111-11-11
02	게시판 제목이 표출됩니다.	2111-11-11
01	게시판 제목이 표출됩니다.	2111-11-11

« « 1 · 2 · 3 · 4 · 5 » »

등록



Welcome to Korea

[Home](#)

[board](#)

[exchange rate report](#)

[sign up](#)

[login](#)

2019.11.22 20:00 KEB하나은행 기준 고시환율 321회

	currency	basic rate	cash(buy)	cash(sell)	remittance(send)	remittance(get)	conversion with USD
0	미국USD	1178.00	1198.61	1157.39	1189.50	1166.50	1.000
1	유럽연합EUR	1302.40	1328.31	1276.49	1315.42	1289.38	1.106
2	일본JPY(100엔)	1085.36	1104.35	1066.37	1095.99	1074.73	0.921
3	중국CNY	167.31	175.67	158.95	168.98	165.64	0.142
4	홍콩HKD	150.55	153.51	147.59	152.05	149.05	0.128
5	대만TWD	38.57	43.62	35.88	0.00	0.00	0.033
6	영국GBP	1517.32	1547.21	1487.43	1532.49	1502.15	1.288
7	오만OMR	3059.74	3332.05	2876.16	0.00	0.00	2.597
8	캐나다CAD	886.81	904.28	869.34	895.67	877.95	0.753
9	스위스CHF	1184.63	1207.96	1161.30	1196.47	1172.79	1.006
10	스웨덴SEK	122.28	125.27	119.29	123.50	121.06	0.104
11	호주AUD	799.74	815.49	783.99	807.73	791.75	0.679
12	뉴질랜드NZD	755.98	770.87	741.09	763.53	748.43	0.642
13	체코CZK	51.03	55.36	46.44	51.59	50.47	0.043
14	칠레CLP	1.48	1.59	1.37	0.00	0.00	0.001
15	터키TRY	206.76	0.00	0.00	209.03	204.49	0.176
16	몽골MNT	0.44	0.00	0.00	0.00	0.00	0.000
17	이스라엘ILS	340.02	374.02	312.82	0.00	0.00	0.289
18	덴마크DKK	174.28	178.54	170.02	176.02	172.54	0.148
19	노르웨이NOK	128.62	131.77	125.47	129.90	127.34	0.109
20	사우디아라비아SAR	314.12	333.90	292.45	317.26	310.98	0.267
21	쿠웨이트KWD	3878.19	4130.27	3567.94	3916.97	3839.41	3.292
22	바레인BHD	3124.25	3324.20	2874.31	3155.49	3093.01	2.652
23	아랍에미리트AED	320.73	338.37	298.60	323.93	317.53	0.272
24	요르단JOD	1661.50	1809.37	1528.58	0.00	0.00	1.410
25	이집트EGP	73.12	0.00	0.00	0.00	0.00	0.062
26	태국THB	39.03	40.98	36.69	39.42	38.64	0.033
27	시리아SYP	664.07	661.57	647.47	670.61	655.70	0.721

02. 개요



(서울) 외국인 거주 정보 제공

1. 서울 행정구역(구) 당 나라별, 연령별, 비자별
외국인 현황
2. 각 나라별로 한국에 방문하는 월별 추이
월별 추이를 통해서 다음 년도 방문 예측



- 회원가입
 - 게시판을 이용하기 위한 간단한 형식
- 나라별 커뮤니티 게시판
 - 각 나라별 자국어로 정보 공유 가능한 게시판
 - 참고(타임포럼 – 브랜드 게시판)

전체	Damasko	Glashütte Original	Junghans	Meistersinger	Stowa	Sinn	Steinhart	NOMOS	Zeppelin	ETC(기타브랜드)	공지사항			
번호	제목										글쓴이	날짜	조회 수	추천 수
공지	독일 시계 브랜드 일람 [127]										hayan	2010.09.06	16990	43
Hot	드디어 받았습니다! [24]										민 촉영	2019.07.25	1318	6
Hot	참지 못하고 결국 질렀습니다. [12]										민 촉영	2019.07.25	1255	2
Hot	1년을 망설이다.. 입당했어요^^/Seventies [14]										enjoywatch	2019.06.18	1389	4
Hot	856 세팅 미감 한정판 [16]										마근엄	2019.03.24	1116	8
8459	[NOMOS] 노모스 탕센테 139 [1]	■■■■■									간고등어	2019.11.03	100	1
8458	[Junghans] 출근길 막스빌 [6]	■■■■■									밍구1	2019.11.01	148	2

서론-경험 기술, 개발도구

및 구축환경



OvenApp.io



draw.io



B Bootstrap

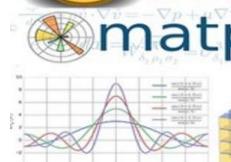


django



pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



matplotlib



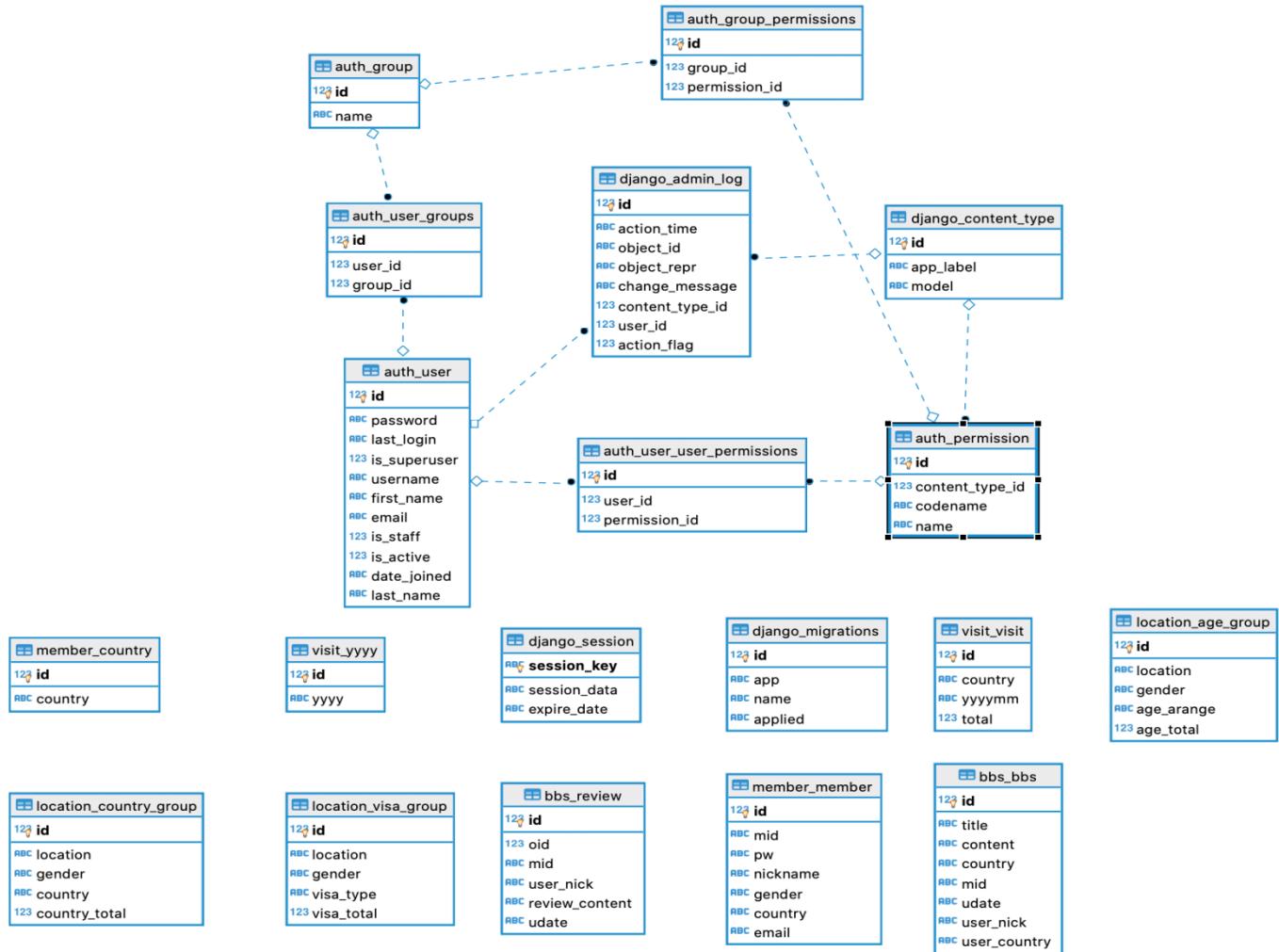
MySQL
NumPy



eSoftner

Google Developers Charts





3. DB설계



Member_member

속성	속성타입	제약조건	내용
mid	Int	Not null, Auto in	아이디
pw	Varchar	Not Null	비밀번호
nickname	Varchar	Not Null	닉네임
gender	Varchar	Not Null	성별
country	Varchar	Not Null	나라
email	Varchar	Not Null	이메일

3. DB설계



Member_country

속성	속성타입	제약조건	내용
id	Integer	Not null, Auto in	아이디
country	Varchar	Not Null	나라

3. DB설계



Bbs_bbs

속성	속성타입	제약조건	내용
id	Integer	Auto Increment	아이디
title	Varchar	Not Null	제목
content	Varchar	Not Null	콘텐트
country	Varchar	Not Null	나라
mid	Varchar	Not Null	테이블 아이디
update	Datetime	Not Null	작성시간
User_nick	Varchar		유저 닉네임
User_country	Varchar		유저 나라

3. DB설계

Bbs_review



속성	속성타입	제약조건	내용
id	Integer	Auto Increment, pk	아이디
Oid	Integer	Not Null	게시판 아이디
Mid	Varchar		닉네임
User_nick	Varchar	Not Null	나라
Review_content	Text	Not Null	댓글
update	Datetime	Not Null	작성시간

3. DB설계



Visit_visit

속성	속성타입	제약조건	내용
id	Intetger	Auto Increment PK	아이디
country	Varchar	Not Null	나라
yyymm	Varchar	Not Null	년도
total	Integer	Not Null	총 방문수

3. DB설계



Visit_yyyy

속성	속성타입	제약조건	내용
id	Integer	Auto Increment PK	아이디
yyyy	Varchar	Not Null	년도

3. DB설계



Location_age_group

속성	속성타입	제약조건	내용
id	Integer	Auto Increment PK	아이디
Location	Varchar	Not Null	지역구
Gender	Varchar	Not Null	성별
Age_arrange	Varchar	Not Null	나이
Age_total	Integer	Not null	같은 나이의 총수

3. DB설계



Location_country_group

속성	속성타입	제약조건	내용
id	Integer	Auto Increment PK	아이디
Location	Varchar	Not Null	지역구
Gender	Varchar	Not Null	성별
County	Varchar	Not Null	나라
Country_toal	Integer	Not null	나라의 총수

3. DB설계



Location_visa_group

속성	속성타입	제약조건	내용
id	Integer	Auto Increment PK	아이디
Location	Varchar	Not Null	지역구
Gender	Varchar	Not Null	성별
Visa_type	Varchar	Not Null	비자 종류
Visa_total	Integer	Not null	비자의 총 합계

04. 기술 설명(김갑동)



```
class Country_group(models.Model):
    location = models.CharField(max_length=100)
    gender = models.CharField(max_length=100)
    country = models.CharField(max_length=100)
    country_total = models.IntegerField(default=0)

    def __str__(self):
        return self.location + ', ' + self.gender + ', ' + self.country + ', ' + str(self.country_total)
```

```
class Visa_group(models.Model):
    location = models.CharField(max_length=100)
    gender = models.CharField(max_length=100)
    visa_type = models.CharField(max_length=100)
    visa_total = models.IntegerField(default=0)

    def __str__(self):
        return self.location + ', ' + self.gender + ', ' + self.visa_type + ', ' + str(self.visa_total)
```

```
class Age_group(models.Model):
    location = models.CharField(max_length=100)
    gender = models.CharField(max_length=100)
    age_arange = models.CharField(max_length=100)
```

1	강남구	남	한국계중국인	294
2	강남구	남	중국	598
3	강남구	남	베트남	68
4	강남구	남	우즈베키스탄	23
5	강남구	남	캄보디아	7
6	강남구	남	필리핀	16
7	강남구	남	인도네시아	3
8	강남구	남	네팔	16
9	강남구	남	태국	33
10	강남구	남	스리랑카	1
11	강남구	남	미국	452

DB에서 테이블을 먼저 설정해 줄 수 있다.

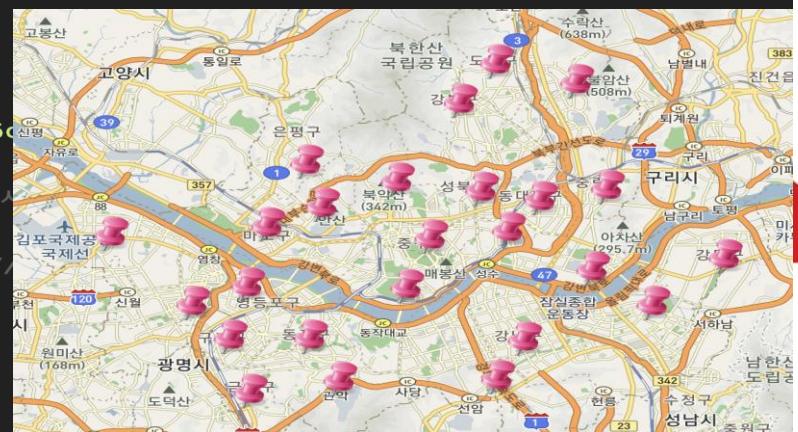
120 id	RBC location	RBC gender	RBC age_arange	123 age_total
1	강남구	남	from0to4	50
2	강남구	남	from5to9	41
3	강남구	남	from10to14	35
4	강남구	남	from15to19	46
5	강남구	남	from20to24	142
6	강남구	남	from25to29	405
7	강남구	남	from30to34	533
8	강남구	남	from35to39	350
9	강남구	남	from40to44	249

```

{%
block script2 %}

<div id="map" style="..."></div>
<script src="/static/jquery-3.4.1.js"></script>
<script src="//dapi.kakao.com/v2/maps/sdk.js?appkey=b4ec527c456f6c9e433352a2f3078d3"></script>
<script>
    var mapContainer = document.getElementById('map'), // 지도를 표시할 캐비닛
        mapOption = {
            center: new kakao.maps.LatLng(37.54130, 126.98267), // 지도의 중심 좌표
            level: 9, // 지도의 확대 레벨
            mapTypeId: kakao.maps.MapTypeId.ROADMAP // 지도종류
        };
    // 지도를 생성한다
    var map = new kakao.maps.Map(mapContainer, mapOption);
    // 마우스 드래그와 모바일 터치를 이용한 지도 이동을 막는다
    map.setDraggable(false);
    // 마우스 휠과 모바일 터치를 이용한 지도 확대, 축소를 막는다
    map.setZoomable(false);
    // 지도 영역 변화 이벤트를 등록한다
    kakao.maps.event.addListener(map, 'bounds_changed', function () {
        var mapBounds = map.getBounds(),
            message = '지도의 남서쪽, 북동쪽 영역좌표는 ' +
            mapBounds.toString() + '입니다.';
    });
    var positions = [
        {

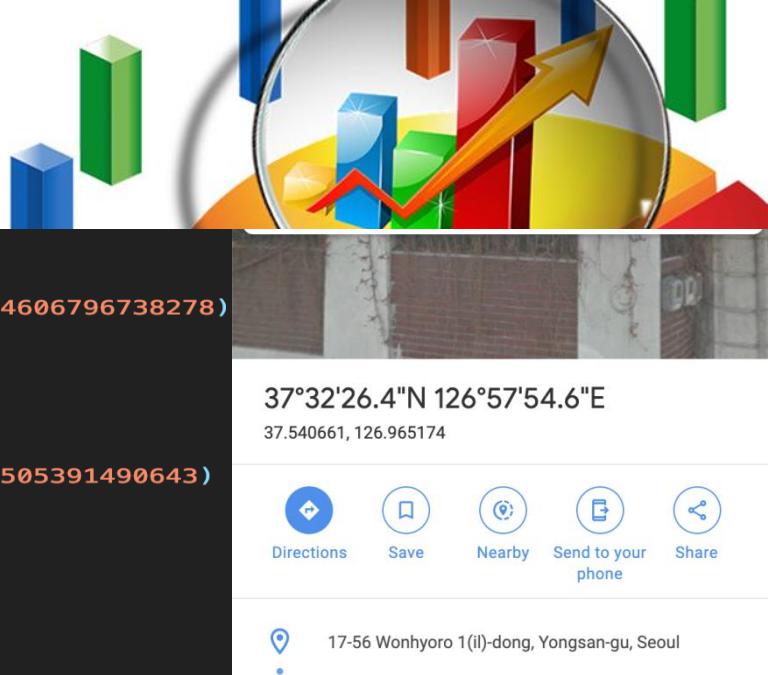
```



카카오 디벨로퍼 도메인 등록 API key 값 설정

JS 으로 HTML에서 화면 구현

```
        title: '강남구',
        content: '<div style="padding:5px;">강남구</div>',
        latlng: new kakao.maps.LatLng(37.496881433189756, 127.04606796738278)
    },
    {
        title: '강동구',
        content: '<div style="padding:5px;">강동구</div>',
        latlng: new kakao.maps.LatLng(37.54764120657503, 127.15505391490643)
    },
    {
        title: '강북구',
        content: '<div style="padding:5px;">강북구</div>',
        latlng: new kakao.maps.LatLng(37.643390, 127.011767)
    },
    {
        title: '강서구',
        content: '<div style="padding:5px;">강서구</div>',
        latlng: new kakao.maps.LatLng(37.561564, 126.822603)
    },
    {
        title: '관악구',
        content: '<div style="padding:5px;">관악구</div>',
        latlng: new kakao.maps.LatLng(37.472082582273266, 126.94657090508751)
    },
    {
        title: '관악구'
```



구마다 좌표를 검색후 입력



```
        title: positions[i].title,
        image: markerImage
    });
    var infowindow = new kakao.maps.InfoWindow({
        content: positions[i].content // 인포윈도우에 표시할 내용
    });

    kakao.maps.event.addListener(marker, 'click', function(mouseEvent){
        location.href="/location/"+this.mc
    });
    kakao.maps.event.addListener(marker, 'mouseover', makeOverListener(map, marker, infowindow));
    kakao.maps.event.addListener(marker, 'mouseout', makeOutListener(infowindow));
}
function makeOverListener(map, marker, infowindow) {
    return function () {
        infowindow.open(map, marker);
    };
}
function makeOutListener(infowindow) {
    return function() {
        infowindow.close();
    };
}
</script>
{% endblock %}
```

1. 클릭시 설정 좌표의 구로 이동 할 수
있도록 href 설정

2. 마우스 클릭 시 marker makeover or out

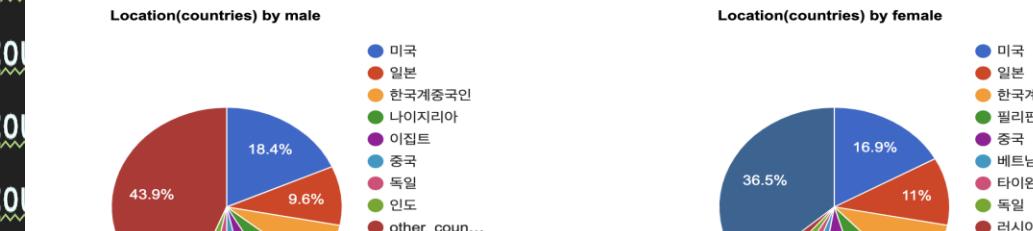
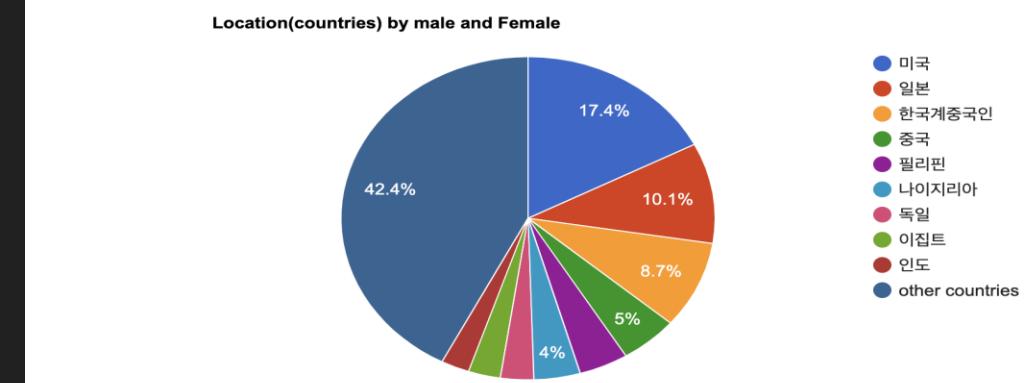
```
google.charts.load('current', {'packages':['corechart']});  
google.charts.setOnLoadCallback(drawChart0);
```

```
function drawChart0() {
```

```
var data0 = new google.visualization.DataTable();  
data0.addColumn('string', 'country');  
data0.addColumn('number', 'country_total');  
data0.addRows([  
  
    '{{total_data.0.country}}', {{total_data.0.co...  
    '{{total_data.1.country}}', {{total_data.1.co...  
    '{{total_data.2.country}}', {{total_data.2.co...  
    '{{total_data.3.country}}', {{total_data.3.co...
```

1. Google.charts 사용 View 처리한
data 표현하기

2. Inside JS data 값을 insert





```
def country_group(request, location):
    total_data = Country_group.objects.filter(location=location).values('country').annotate(Sum('country_total')).order_by('-country_total')
    gender_m = Country_group.objects.filter(location=location, gender='남').order_by('-country_total')
    gender_f = Country_group.objects.filter(location=location, gender='여').order_by('-country_total')

    total_countries = 0

    for j in range(9, len(total_data)):
        total_countries = total_countries + total_data[j]['country_total__sum']

    gender_m_sum=0
    gender_f_sum=0

    for i in range(9, len(gender_m)):
        gender_m_sum = gender_m_sum + gender_m[i].country_total
        gender_f_sum = gender_f_sum + gender_f[i].country_total

    country_list = Country.objects.all().order_by('country')

    list_gender = {'gender_m':gender_m, 'gender_f':gender_f, 'gender_f_sum':gender_f_sum, 'gender_m_sum':gender_m_sum, 'total_data':total_data}

    return render(request, 'country_group/country_group.html', list_gender)
```

1. Import Country_group from DB 후 전처리

2. Sum, order by, 오름차순, for 문, 전처리를 통해서 원하는 데이터를 함수로 설정

```

chart0.draw(data0, options0);

var selectHandler = function () {
    var selectedItem = chart0.getSelection()[0];
    var value = data0.getValue(selectedItem.row, 0);
    if (value == "other countries") {
    } else {
        location.href = "/visit/?country=" + value + "&yyyy=2017"
    }
};

google.visualization.events.addListener(drawChart, "ready");

$(document).ready(function () {
    $('#btn').click(function () {
        drawChart();
    });
});

```

1. href 클릭시 다음 server 으로 갈 수 있도록 설정.

2. If 문으로 other countries 는 이동 하지 않게 사용.



정병수(상속)



Dstagram Home board exchange rate report sign up login

상속해줄
Base.html 작성 후

순번	제목	작성자	날짜
4	你好,这里真的很好	nickname1 한국	Nov. 22, 2019, 8:49 a.m.
3	こんにちは。ここ大好きです。	nickname1 한국	Nov. 22, 2019, 8:48 a.m.
2	title2	nickname2 네덜란드	Nov. 19, 2019, 4:36 p.m.
1	title1	nickname1 한국	Nov. 19, 2019, 4:36 p.m.

각 template에서
상속 받아서
화면구성

정병수/ajax-id중복확인



ID	<input type="text" value="mid1"/>	중복된 아이디입니다
PW	<input type="password"/>	
PW Confirm	<input type="password"/>	
Nickname	<input type="text"/>	
Gender	남 ▼	
Country	한국 ▼	
Email	<input type="text"/>	
<input type="button" value="가입"/>		

ID	<input type="text" value="mid3"/>	사용가능한 아이디입니다
PW	<input type="password"/>	
PW Confirm	<input type="password"/>	
Nickname	<input type="text"/>	
Gender	남 ▼	
Country	한국 ▼	
Email	<input type="text"/>	
<input type="button" value="가입"/>		

<Ajax 활용>
회원가입 시
id중복확인 후
keyup할 때마다
Template에 결과
리턴

정병수(crawling)



2019.11.22 20:00 KEB하나은행 기준 고시환차321회

	currency	basic rate	cash(buy)	cash(sell)	remittance(send)	remittance(get)	conversion with USD
0	미국USD	1178.00	1198.61	1157.39	1189.50	1166.50	1.000
1	유럽연합EUR	1302.40	1328.31	1276.49	1315.42	1289.38	1.106
2	일본JPY(100엔)	1085.36	1104.35	1066.37	1095.99	1074.73	0.921
3	중국CNY	167.31	175.67	158.95	168.98	165.64	0.142
4	홍콩HKD	150.55	153.51	147.59	152.05	149.05	0.128
5	대만TWD	38.57	43.62	35.88	0.00	0.00	0.033
6	영국GBP	1,517.32	1,547.21	1,487.43	1,532.49	1,502.15	1.288
7	오만OMR	3,059.74	3,332.05	2,876.16	0.00	0.00	2.597
8	캐나다CAD	886.81	904.28	869.34	895.67	877.95	0.753
9	스위스CHF	1,184.63	1,207.96	1,161.30	1,196.47	1,172.79	1.006
10	스웨덴SEK	122.28	125.27	119.29	123.50	121.06	0.104
11	호주AUD	799.74	815.49	783.99	807.73	791.75	0.679
12	뉴질랜드NZD	755.98	770.87	741.09	763.53	748.43	0.642
13	체코CZK	51.03	55.36	46.44	51.59	50.47	0.043
14	칠레CLP	1.48	1.59	1.37	0.00	0.00	0.001
15	터키TRY	206.76	0.00	0.00	209.03	204.49	0.176
16	몽골MNT	0.44	0.00	0.00	0.00	0.000	
17	이스라엘ILS	340.02	374.02	312.82	0.00	0.00	0.289
18	엔마크DKK	174.28	178.54	170.02	176.02	172.54	0.148
19	노르웨이NOK	128.62	131.77	125.47	129.90	127.34	0.109
20	사우디아라비아SAR	314.12	333.90	292.45	317.26	310.98	0.267
21	쿠웨이트KWD	3,878.19	4,130.27	3,567.94	3,916.97	3,839.41	3.292

통화명	매매기준율	환찰		송금		미화환산율
		사실 때	파실 때	보내실 때	받으실 때	
미국 USD	1,178.00	1,198.61	1,157.39	1,189.50	1,166.50	1.000
유럽연합 EUR	1,302.40	1,328.31	1,276.49	1,315.42	1,289.38	1.106
일본 JPY (100엔)	1,085.36	1,104.35	1,066.37	1,095.99	1,074.73	0.921
중국 CNY	167.31	175.67	158.95	168.98	165.64	0.142
홍콩 HKD	150.55	153.51	147.59	152.05	149.05	0.128
대만 TWD	38.57	43.62	35.88	0.00	0.00	0.033
영국 GBP	1,517.32	1,547.21	1,487.43	1,532.49	1,502.15	1.288
오만 OMR	3,059.74	3,332.05	2,876.16	0.00	0.00	2.597
캐나다 CAD	886.81	904.28	869.34	895.67	877.95	0.753
스위스 CHF	1,184.63	1,207.96	1,161.30	1,196.47	1,172.79	1.006
스웨덴 SEK	122.28	125.27	119.29	123.50	121.06	0.104
호주 AUD	799.74	815.49	783.99	807.73	791.75	0.679
뉴질랜드 NZD	755.98	770.87	741.09	763.53	748.43	0.642
체코 CZK	51.03	55.36	46.44	51.59	50.47	0.043
칠레 CLP	1.48	1.59	1.37	0.00	0.00	0.001

<crawling 활용>
페이지 이동 시
실시간 환율정보
출력

정병수(댓글구현)



update	delete				
title1					
nickname1	일본	Nov. 19, 2019, 4:36 p.m.			
content1					
comment					
NickName	nickname1	Update	Nov. 24, 2019, 9:33 a.m.	Update	Delete
comment15					
NickName	nickname2	Update	Nov. 20, 2019, 2:45 a.m.		

update	delete				
title1					
nickname1	일본	Nov. 19, 2019, 4:36 p.m.			
content1					
comment					
NickName	nickname1	Update	Nov. 24, 2019, 9:33 a.m.	Done	Cancel
comment15					

<flag 활용>
게시글 상세내용
페이지 이동 시
또는 redirect 시
Flag값을 넘겨
읽기전용 ->
수정가능 댓글로
변경하기

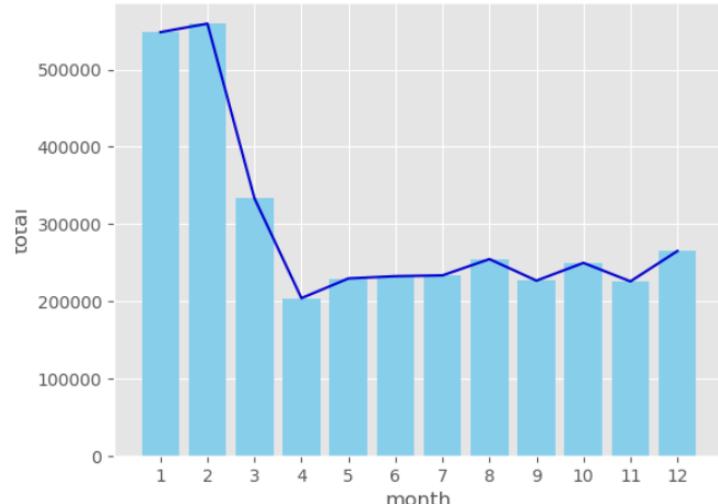
04. 기술 설명 (민희원)

데이터 시각화

```
def visit(request):
    data = Visit.objects.filter(country=request.GET['country'],
yyyymm__startswith=request.GET['yyyy'])
    list1 = []
    list2 = []
    for i in data:
        list1.append(i.total)
        list2.append(i.yyyymm.replace('월','').replace(
request.GET['yyyy']+'.',''))
    df = pd.DataFrame({'month':list2,'total':list1})
    df_table = df.to_html()
    matplotlib.use('Agg')
    matplotlib.style.use('ggplot')
    plt.xlabel('month')
    plt.ylabel('total')
    plt.bar(list2, list1, color='skyblue')
    plt.plot(list2, list1, color='mediumblue')
    plt.savefig('visit/static/graph.png')
    plt.close()
```



중국 ▾ 2017 ▾ 이동 예측



웹 연결

(feat.form태그)

```
<script>
    $(function(){
        $("#predict").click(function(){
            var country = $("#country").val()
            location.href="/visit/predict?country="+country
        })
    })
</script>
#...생략
<form action="/visit/">
    <select name="country" id="country">
        <option value="{{list.country}}">{{list.country}}</option>
        {% for country in list2 %}
            <option value="{{country.country}}">{{country.country}}</option>
        {% endfor %}
    </select>
    <select name="yyyy">
        <option value="{{list4}}" selected>{{list4}}</option>
        {% for yyyy in list3 %}
            <option value="{{yyyy.yyyy}}">{{yyyy.yyyy}}</option>
        {% endfor %}
    </select>
    <input type="submit" value="이동">
    <button type="button" id="predict">예측</button>
</form>
</td>
</tr>
<tr>
    <td align="center"></td>
</tr>
<tr>
    <td align="center"></td>
</tr>
</table>
```

Forecasting 예측 (feat. Python.fbprophet)



```
In [15]: import pandas as pd  
import numpy as np  
from fbprophet import Prophet
```

```
In [16]: df2 = pd.read_csv('C:/Users/Administrator/Desktop/data/cntry/china.csv', parse_dates=[0])
```

```
In [17]: df2.head(n=10)
```

```
Out [17]:
```

	ds	y
0	1998-01-01	13348
1	1998-02-01	13562
2	1998-03-01	15128
3	1998-04-01	13880
4	1998-05-01	16095
5	1998-06-01	15751
6	1998-07-01	16021
7	1998-08-01	18027
8	1998-09-01	17951
9	1998-10-01	18920

```
In [18]: df2.dtypes
```

```
Out [18]:
```

ds	datetime64[ns]
y	int64
dtype:	object

```
In [19]:
```

```
import warnings  
warnings.filterwarnings("ignore")  
import pandas as pd  
import pandas_datareader.data as web  
import numpy as np  
import matplotlib.pyplot as plt  
%matplotlib inline  
  
from fbprophet import Prophet  
from datetime import datetime
```

```
In [20]:
```

```
path = "c:/Windows/Fonts/malgun.ttf"  
import platform  
from matplotlib import font_manager, rc  
if platform.system() == 'Darwin':  
    rc('font', family='AppleGothic')  
elif platform.system() == 'Windows':  
    font_name = font_manager.FontProperties(fname=path).get_name()  
    rc('font', family=font_name)  
else:  
    print('Unknown system... sorry~~~~')  
  
plt.rcParams['axes.unicode_minus'] = False
```

Forecasting 예측 (feat. Python.fbprophet)

*NaN값 있으면 drop 해주기

```
In [23]: m = Prophet()
```

```
In [24]: m.fit(df2)
```

```
INFO:fbprophet:Disabling weekly seasonality. Run prophet with weekly_seasonality=True to override this.  
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to override this.
```

```
Out [24]: <fbprophet.forecaster.Prophet at 0x170de8cd608>
```

```
In [25]: future = m.make_future_dataframe(periods=365)  
future.tail()
```

```
Out [25]:
```

ds
611 2019-11-27
612 2019-11-28
613 2019-11-29
614 2019-11-30
615 2019-12-01

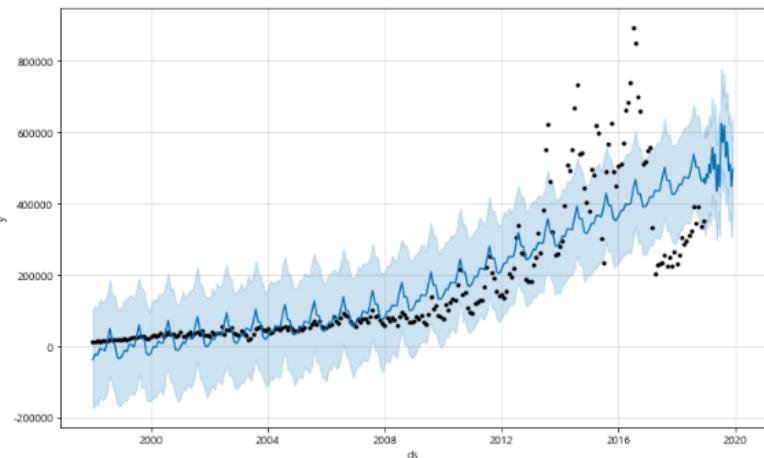
```
In [26]: forecast = m.predict(future)  
forecast[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].tail()
```

```
Out [26]:
```

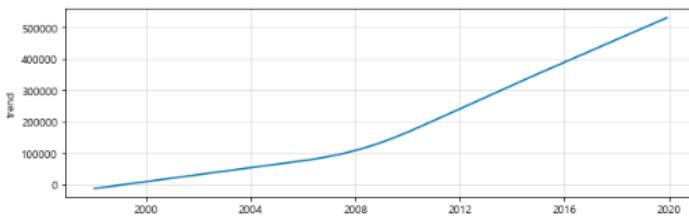
ds	yhat	yhat_lower	yhat_upper
611 2019-11-27	480978.844074	347852.757636	619122.198014
612 2019-11-28	485454.667601	359402.345218	625960.614344
613 2019-11-29	489749.508858	359899.104888	633259.190455
614 2019-11-30	493769.917082	360049.727857	635484.176414
615 2019-12-01	497433.045765	364849.925473	626229.059960



```
In [27]: fig1 = m.plot(forecast)
```



```
In [28]: fig2 = m.plot_components(forecast)
```



Djang 웹 연결 #hidden page

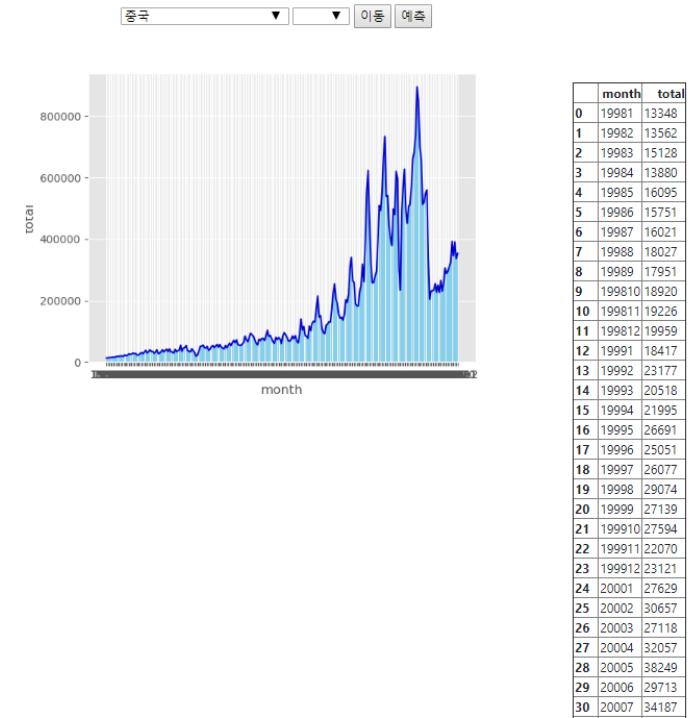
```
def predict(request):
    data = request.GET
    pic_type1 = '/static/' + data['country'] + '_line+seasonal.png'
    pic_type2 = '/static/' + data['country'] + '_whole.png'
    data2 = Country.objects.all().order_by('country')
    data3 = Yyyy.objects.all().order_by('-yyyy')
    pic_url =
{'pic1':pic_type1,'pic2':pic_type2,'list':data,'list2':data2,'list3':data3}
    return render(request, 'visit/predict.html',pic_url)

from django.shortcuts import render
# Create your views here.
def visit(request):
    data =
Visit.objects.filter(country=request.GET['country'],
yyyymm__startswith=request.GET['yyyy'])
#~~~생략 (as aforementioned)

    list = {'list': data, 'list2':data2,
'list3':data3,'list4':request.GET['yyyy'],
'img_url':img_url, 'df_table':df_table}
    return render(request, 'visit/visit.html', list)
```



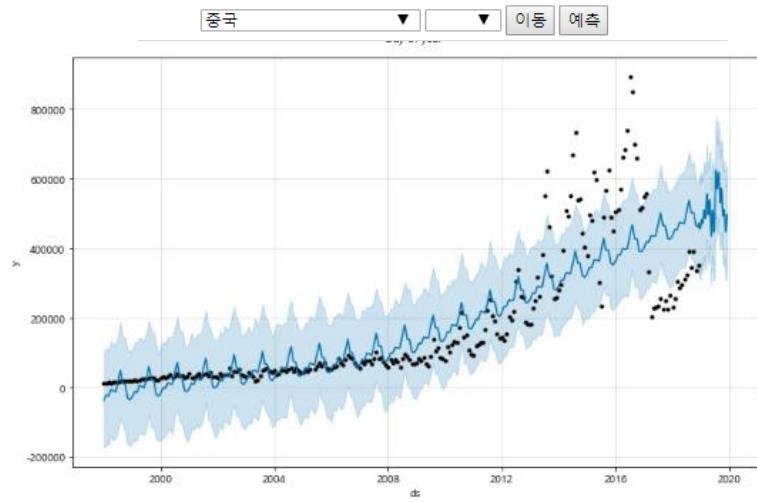
visit page



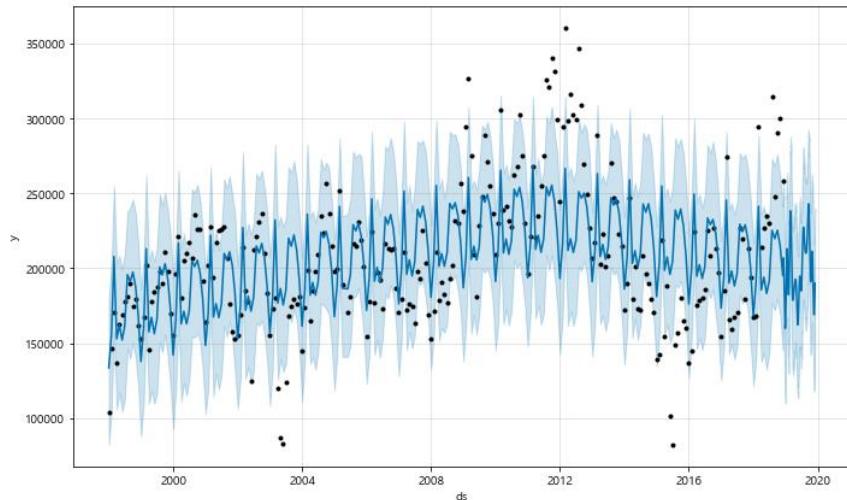
예측(predict)페이지를 갔다오면 'yyyy'값은 blank가 되지만 'country' 값은 그대로



predict page



Vs. 일본





- 모두 고생 많으셨습니다. 특히 우리 팀 끝까지
마무리 잘 해주셔서 감사합니다.