커스텀 유저 Admin 만들기

(중급편) 장고 기본 인증 뽀개기

Ask Company

커스텀 User Manager

accounts/models.py

```python
from django.contrib.auth.models import UserManager as AuthUserManager

class UserManager(AuthUserManager):
    def create_superuser(self, username, email, password, **extra_fields):
        extra_fields.setdefault('sex', 'm')  # sex 필드에 대한 디폴트값 지정
        return super().create_superuser(username, email, password, **extra_fields)

class User(AbstractUser):
    sex = models.CharField(
            max_length=1,
            choices=(
                ('f', 'female'),
                ('m', 'male'),
            ))

    objects = UserManager()
```

커스텀 User Admin

accounts/admin.py

```python
from django.contrib.auth.admin import UserAdmin as AuthUserAdmin
from .models import User
from .filters import UserDateJoinedFilter


@admin.register(User)
class UserAdmin(AuthUserAdmin):
    list_display = ('username', 'email', 'first_name', 'last_name', 'is_staff')
    list_filter = ('is_staff', 'is_superuser', 'is_active', 'groups', 'date_joined', 'sex', UserDateJoinedFilter)
    search_fields = ('username', 'first_name', 'last_name', 'email')
    actions = ['마케팅_이메일보내기']

    def 마케팅_이메일보내기(self, request, queryset):
        for user in queryset:
            pass
        self.message_user(request, 'hello world')
```

커스텀 User Admin을 위한 커스텀 필터

accounts/filters.py

```python
from django.contrib import admin

class UserDateJoinedFilter(admin.SimpleListFilter):
    title = '유저 가입일'
    parameter_name = 'date_joined__match'

    def lookups(self, request, model_admin):
        candidate = []

        # 현재 월의
        start_date = timezone.now().replace(day=1, hour=0, minute=0, second=0, microsecond=0)

        for i in range(6):
            value = '{}-{}'.format(start_date.year, start_date.month)
            label = '{}년 {}월 가입자'.format(start_date.year, start_date.month)
            candidate.append([value, label])
            start_date -= relativedelta(months=1)

        return candidate

    def queryset(self, request, queryset):
        value = self.value()

        if not value:
            return queryset

        try:
            year, month = map(int, value.split('-'))
            queryset = queryset.filter(date_joined__year=year, date_joined__month=month)
        except ValueError:
            return queryset.none()

        return queryset
```

Life is short,
Use Python3/Django.

여러분의
파이썬/장고 페이스메이커가
되겠습니다. ;)