地理编码和反向地理编码

地理编码指的是找出人类能够看懂的地址转换为对应的经度和纬度坐标；反向地理编码指的是根据坐标翻译成人类能看得懂的地址。编码器CLGeocoder的每个实例，同时只能处理一个任务，异步执行。

对地址进行地理编码

根据地理位置信息，用CLGeocoder实例编码为地理坐标信息，因为文字内容的不确定性，可能匹配多个地方。

添加用户输入内容事件

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    [self.textFild addTarget:self action:@selector(editTextEnd:) forContr
olEvents:UIControlEventEditingDidEndOnExit];
    self.textFild.returnKeyType = UIReturnKeyDone;
}
```

CLGeocoder

根据用户输入的地理位置信息，用CLGeocoder实例编译成地理坐标信息，

CLPlacemark

用MKPointAnnotation在地图上标出,下面代码考虑了如何让多个点在屏幕上合适的显示

```
-(void)editTextEnd:(UITextField *)textField{
    NSLog(@">>>>>%@", textField.text);

    CLGeocoder *geocodr = [[CLGeocoder alloc] init];
    [geocodr geocodeAddressString:textField.text completionHandler:^(NSAr
ray *placemarks, NSError *error) {

        [self.mapView removeAnnotations:self.mapView.annotations];

        double minLatitude = 90.f;
        double maxLatitude = -90.f;
        double minLongitude = 180.f;
        double maxLongitude = -180.f;
```

```
        for (CLPlacemark *placemark in placemarks) {
            NSLog(@"%@", placemark.name);
            CLLocation *location = [placemark location];

            MKPointAnnotation *ann = [[MKPointAnnotation alloc] init];
            ann.coordinate = location.coordinate;
            ann.title = placemark.name;
            [self.mapView addAnnotation:ann];


            //找到所有点经纬度的最大最小值
            minLatitude = location.coordinate.latitude < minLatitude ? lo
cation.coordinate.latitude : minLatitude;
            maxLatitude = location.coordinate.latitude > maxLatitude ? lo
cation.coordinate.latitude : maxLatitude;

            minLongitude = location.coordinate.longitude < minLongitude ?
 location.coordinate.longitude : minLongitude;
            maxLongitude = location.coordinate.longitude > maxLatitude ?
location.coordinate.longitude : maxLatitude;

        }


//        计算区域
        double centerLatitude = (minLatitude + maxLatitude)/2;
        double centerLongitude = (minLongitude + maxLongitude)/2;
        double spanLatitude = maxLatitude - minLatitude;
        double spanLongitude = maxLongitude - minLongitude;

        CLLocationCoordinate2D coordinate = CLLocationCoordinate2DMake(ce
nterLatitude, centerLongitude);

        [self.mapView setRegion:MKCoordinateRegionMake(coordinate, MKCoor
dinateSpanMake(spanLatitude * 1.5, spanLongitude * 1.5)) animated:YES];

    }];

}
```

对位置进行反向地理编码

根据得到的经纬度坐标， 通过CLGeocoder反向的编码得到位置信息

添加长按手势，选择位置

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    UILongPressGestureRecognizer *longPress = [[UILongPressGestureRecogni
zer alloc] initWithTarget:self action:@selector(longPress:)];
    [self.mapView addGestureRecognizer:longPress];
    self.longPress = longPress;

    //为方便用户选择，用mapView定位到当前位置，并放大区域
    self.mapView.showsUserLocation = YES;
    self.mapView.delegate = self;
}

-(void)mapView:(MKMapView *)mapView didUpdateUserLocation:(MKUserLocation
 *)userLocation{
    [self.mapView setRegion:MKCoordinateRegionMake(userLocation.coordinat
e, MKCoordinateSpanMake(0.05, 0.05)) animated:YES];
}
```

长按手势触发后，将视图坐标转化为地理坐标，然后将地理坐标翻译为地理位置信息

```
-(void)longPress:(UIGestureRecognizer *)press{
    CGPoint point = [press locationInView:self .mapView];
    NSLog(@"%@", NSStringFromCGPoint(point));

    CLLocationCoordinate2D location = [self.mapView convertPoint:point to
CoordinateFromView:self.mapView];
    CLLocation *cl = [[CLLocation alloc] initWithLatitude:location.latitu
de longitude:location.longitude];

    CLGeocoder *geocodr = [[CLGeocoder alloc] init];
    [geocodr reverseGeocodeLocation:cl completionHandler:^(NSArray *place
marks, NSError *error) {
        NSLog(@"%@", placemarks.firstObject);
        MKPointAnnotation *annotation = [[MKPointAnnotation alloc] init];
        MKPlacemark *placemark = [placemarks objectAtIndex:0];
        CLLocation *location = placemark.location;
        annotation.coordinate = location.coordinate;
        annotation.subtitle = placemark.name;
        annotation.title = placemark.country;
        [self.mapView addAnnotation:annotation];
    }];

}
```