

マップの使い方

- 1.はじめに映す視野確保
- 2.緯度経度の設定
- 3.傾きの設定
- 4.デリゲートの設定
- 5.mapへ設定視野をセットすつ

マップの使い方

- ① `MKCoordinateRegion region = {{0.0,0.0},{0.0,0.0}}`
- ② `region.center.latitude = 35.624129;`
- ② `region.center.longitude = 139.733017;`
- ③ `region.span.latitudeDelta = 0.01f;`
- ③ `region.span.longitudeDelta = 0.01f;`
- ④ `map.delegate = self;`
- ⑤ `[map setRegion:region animated:YES];`

マップの使い方



1. ピンの初期化
2. ピンが打たれる緯度経度の設定
3. ピンをタップした時に表示される
タイトルとサブタイトルの設定
4. ピンへタグを設定
5. ピンをマップへ追加

マップの使い方

- ① `MKPointAnnotation *annotation = [[MKPointAnnotation alloc] init];`
- ② `annotation.coordinate = CLLocationCoordinate2DMake(35.624129, 139.733017);`
- ③ `annotation.title = @"お気に入り";`
- ③ `annotation.subtitle = @"レストラン";`
- ④ `tagCount = 1;`

`// ピンを追加`

- ⑤ `[map addAnnotation:annotation];`

マップの使い方

ピンの詳細を決めるデリゲートメソッド

```
- (MKAnnotationView *)mapView:(MKMapView*)mapView
    viewForAnnotation:(id <MKAnnotation>)annotation
{
    // if annotation is user location, return the method
    if ([annotation isKindOfClass:[MKUserLocation class]]) {
        return nil;
    }

    // make annotation view
    static NSString *identifier = @"PlaceAnnotation";
    MKPinAnnotationView *annotationView = (MKPinAnnotationView *)[map dequeueReusableAnnotationViewWithIdentifier:identifier];
    if (!annotationView) {
        annotationView = [[MKPinAnnotationView alloc] initWithAnnotation:annotation reusIdentifier:identifier];
    }
    // set the some properties
    annotationView.pinTintColor = [UIColor greenColor]; //ここはピンの色を決める
    annotationView.canShowCallout = YES;
    annotationView.rightCalloutAccessoryView = [UIButton buttonWithType:UIButtonTypeDetailDisclosure]; //詳細ボタンを設置している
    return annotationView;
}
```

マップの使い方

詳細ボタンタップ時に呼ばれるデリゲートメソッド

```
- (void)mapView:(MKMapView *)mapView annotationView:(MKAnnotationView *)view calloutAccessoryControlTapped:(UIControl *)control {
```

```
    //詳細ボタンがタップされたときに呼ばれるデリゲートメソッド
```

```
}
```