

```
EnrollmentForm
{
  "planNo" : " ",
  "planName" : " ",
  "tenure" : ,
  "issuredAmount" : ,
  "memberName" : " ",
  "dob" : " ",
  "gender" : " ",
  "mobileNo" : " ",
  "emailAddress" : " "
}
```

```
Policy
{
  "policyNo": " ",
  "planName": " ",
  "issuedDate" : " ",
  "effectiveDate" : " ",
  "maturityDate": " ",
  "premiumAmount": " ",
  "tenure" : ,
  "issuedAmount" : ,
  "status" : " "
}
```

```
class EnrollmentForm {
  int planNo;
  String planName;
  int tenure;
  double insurredAmount;
  String memberName;
  LocalDate dob;
  String gender;
  String mobileNo;
  String emailAddress;
  // accessors
}
```

```
class Policy {
  String policyNo;
  String planName;
  LocalDate issuedDate;
  LocalDate effectiveDate;
  LocalDate maturityDate;
  Double premiumAmount;
  int tenure;
  Double insurredAmount;
  String status;
  // accessors
}
```

```
method: POST
URI: /api/policy
headers:
Content-Type: application/json
Body:
{
  "planNo" : " ",
  "planName" : " ",
  "tenure" : ,
  "issuredAmount" : ,
  "memberName" : " ",
  "dob" : " ",
  "gender" : " ",
  "mobileNo" : " ",
  "emailAddress" : " "
}
```



```
@Path("/policy")
class PolicyApiService {
  @POST
  @Consumes(MediaType.APPLICATION_JSON)
  @Produces(MediaType.APPLICATION_JSON)
  public Policy enroll(EnrollmentForm form) {
    // business logic
    return policy;
  }
}
```

```
@Provider
@Produces(MediaType.APPLICATION_JSON)
class JsonMessageBodyWriter implements MessageBodyWriter {
  public boolean isWritable(Class<?> classType, Type
genericType, Annotation[] annotations, MediaType mediaType){
    if(classType == Policy.class) {
      return true;
    }
    return false;
  }

  public void writeTo(Object obj, Class<?> type, Type
genericType, Annotation[] annotations,
    MediaType mediaType,
    MultivaluedMap<String, Object> httpHeaders,
    OutputStream entityStream) {
    JsonB jsonb = JsonBBuilder.create();
    jsonb.toJson(object, entityStream);
    entityStream.close();
  }
}
```

```
@Provider
@Consumes(MediaType.APPLICATION_JSON)
class JsonMessageBodyReader implements MessageBodyReader {

  public boolean isReadable(Class<?> classType, Type genericType,
    Annotation[] annotations, MediaType mediaType) {
    if(classType == EnrollmentForm.class) {
      return true;
    }
    return false;
  }

  public Object readFrom(Class<T> type, Type genericType,
    Annotation[] annotations, MediaType mediaType,
    MultivaluedMap<String, String> httpHeaders,
    InputStream entityStream) {
    Object obj = null;
    JsonB jsonb = JsonbBuilder.createJsonB();
    obj = jsonb.fromJson(entityStream, type);
    return obj;
  }
}
```

```
class JAXBMessageBodyReader implements MessageBodyReader {}
```

```
class JAXBMessageBodyWriter implements MessageBodyWriter {}
```