

Register 기능 수행

👤 이해관계자	
📌 상태	진행 중
👤 작성자	
📌 문서 유형	
🕒 작성일시	@2022년 7월 5일 오후 11:01
🕒 최종 편집일시	@2022년 7월 11일 오전 12:23
👤 최종 편집자	

register 시 필요한 API

npm install body-parser --save

```
//application/x-www-form-urlencoded 를 분석해서 가져옴
app.use(bodyParser.urlencoded({extended:true}));
//application/json형태를 분석해서 가져옴
app.use(bodyParser.json());
```

비밀번호를 평문이 아닌 암호화 된 값을 만들기 위해

bcrypt api 사용

```
const bcrypt = require('bcrypt')
const saltRounds = 10 //10자리인 salt를 생성한 후, 이것을 사용해 암호화
//usermodel의 정보를 저장(save function) 전, pre function 설정
userSchema.pre('save', function(next){
  var user = this; //userSchema 객체

  if(user.isModified('password')){ //model > password 필드가 수정하는 경우
    //bcrypt library로 비밀번호 암호화
    bcrypt.genSalt(saltRounds, function(err, salt){
      if(err) return next(err) //err 나면 save 실행

      bcrypt.hash(user.password, salt, function(err, hash){
        //user.password = 사용자가 입력한 비밀번호 평문, hash = salt를 이용해 평문을 암호화한 값
        if(err) return next(err)
        user.password = hash
        next()
      })
    })
  }
})
```

```
}else{
  next() --> next 는 hash값 생성이 완료 되고 난뒤에 다음 을 수행 하기 위함.
}
})
```

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId("62caed89031a6591cea708c2")
name: "ows"
email: "ows88@naver.com"
password: "$2b$10$6xjfTY1iIz56XsnR0ZLlX.e7m806fpDmwrnNzlfDXHFzHiom6LhT."
__v: 0
```