

MERN Stack Cheat Sheet: JWT, Bcrypt, Cookies, Auth

JSON Web Token (JWT)

Install:

```
npm install jsonwebtoken
```

Import:

```
import jwt from 'jsonwebtoken';
```

Generate Token:

```
const token = jwt.sign({ id: user._id }, process.env.JWT_SECRET, { expiresIn: '7d' });
```

Verify Token:

```
const decoded = jwt.verify(token, process.env.JWT_SECRET);
```

Bcrypt (Password Hashing)

Install:

```
npm install bcrypt
```

Import:

```
import bcrypt from 'bcrypt';
```

Hash Password:

```
const hashedPassword = await bcrypt.hash(password, 10);
```

Compare Passwords:

```
const isMatch = await bcrypt.compare(enteredPassword, user.password);
```

Cookie Handling (Authentication)

Set Cookie:

```
res.cookie('token', token, {  
  httpOnly: true,  
  secure: process.env.NODE_ENV === 'production',  
  sameSite: process.env.NODE_ENV === 'production' ? 'none' : 'strict',  
  maxAge: 7 * 24 * 60 * 60 * 1000, // 7 days  
});
```

Clear Cookie:

```
res.clearCookie('token', { sameSite: ..., secure: ..., httpOnly: true });
```

Auth Middleware Example

```
import jwt from 'jsonwebtoken';
```

MERN Stack Cheat Sheet: JWT, Bcrypt, Cookies, Auth

```
const authMiddleware = (req, res, next) => {
  const token = req.cookies.token;
  if (!token) return res.status(401).json({ message: "Unauthorized" });

  try {
    const decoded = jwt.verify(token, process.env.JWT_SECRET);
    req.userId = decoded.id;
    next();
  } catch (error) {
    res.status(401).json({ message: "Invalid Token" });
  }
};
```

Register/Login Flow (Overview)

Register:

1. Validate input
2. Check if user exists
3. Hash password with bcrypt
4. Save user to DB
5. Generate JWT token and send as cookie

Login:

1. Validate input
2. Find user by email
3. Compare password with bcrypt
4. Generate JWT token and send as cookie

Logout:

1. Clear cookie

MongoDB + Mongoose Basics

Install:

```
npm install mongoose
```

Connect to MongoDB:

```
mongoose.connect(process.env.MONGO_URI, { useNewUrlParser: true, useUnifiedTopology: true });
```

Create Schema & Model:

```
const userSchema = new mongoose.Schema({ name: String, email: String, password: String });
const User = mongoose.model('User', userSchema);
```

Use:

MERN Stack Cheat Sheet: JWT, Bcrypt, Cookies, Auth

```
const user = await User.findOne({ email });  
const newUser = new User({ name, email, password });  
await newUser.save();
```