


## How to use **claim** method in io.jsonwebtoken.JwtBuilder

Best Java code snippets using io.jsonwebtoken.JwtBuilder.claim (Showing top 20 results out of 378)

Common ways to obtain **JwtBuilder**

```
1 private void myMethod () {  
2     JwtBuilder j =  
3 }  
4  
5
```

 **Jwts.builder()**

Smart code suggestions by Tabnine

Get [smart completions](#) for your Java IDE

Add Tabnine to your IDE (free)

JwtToke... *origin: alibaba/nacos*

```
1 /**  
2  * Create token  
3  *  
4  * @param authentication auth info  
5  * @return token  
6  */  
7 public String createToken(Authentic  
8 /**  
9  * Current time  
10 */  
11 long now = (new Date()).getTime()  
12 /**  
13  * Validity date  
14 */  
15 Date validity;  
16 validity = new Date(now + this.to  
17  
18 /**  
19  * create token  
20 */  
21 return Jwts.builder()  
22     .setSubject(authentication.getN  
23     .claim(AUTHORITIES_KEY, "")  
24     .setExpiration(validity)  
25     .signWith(SignatureAlgorithm.HS  
26     .compact();  
27 }  
28
```

J... *origin: SonarSource/sonarqube*

```
5     .setId(uuidFactory.create())
6     .setSubject(jwtSession.getUserLog
7     .setIssuedAt(new Date(now))
8     .setExpiration(new Date(now + jwt
9     .signWith(SIGNATURE_ALGORITHM, se
10    for (Map.Entry<String, Object> ent
11        jwtBuilder. claim(entry.getKey(),
12    }
13    return jwtBuilder.compact();
14 }
15
```

JwtService... *origin: apache/nifi*

```
+
37 .setIssuer(authenticationToken.getI
38 .setAudience(authenticationToken.ge
39 . claim(USERNAME_CLAIM, username)
40 . claim(KEY_ID_CLAIM, key.getId())
41 .setExpiration(expiration.getTime()
42 .setIssuedAt(Calendar.getInstance()
+
```

JwtToke... *origin: mrdear/JavaWEB*

```
1 /**
2  *从用户中创建一个jwt Token
3  *
4  * @param userDTO 用户
5  * @return token
6  */
7 public String create(TokenUserDTO u
8 return Jwts.builder()
9     .setExpiration(new Date(System.c
10    .setSubject(userDTO.getUsername(
11    . claim("id", userDTO.getId())
12    . claim("avatar", userDTO.getAva
13    . claim("email", userDTO.getEmai
14    . claim("roles", userDTO.getRole
15    .signWith(SignatureAlgorithm.HS2
16    .compact();
17 }
18
```

io.jsonwebtoken > JwtBuilder > **claim**

Javadoc

Show more ▾

Authentic... *origin: Exrick/x-boot*

```
+
44 . claim(SecurityConstant.AUTHORITIE
+
```

Popular methods of JwtBuilder

**compact**

Actually builds the JWT and serializes it to a compact, URL-safe stri...

```

5     return Jwts.builder()
6         .setSubject(username)
7         .claim(AUTHORITIES_KEY, authorities)
8         .signWith(SignatureAlgorithm.HS256, secret)
9         .setExpiration(new Date(now + expiration))
10        .compact();
11    }
12

```

... [origin: nielsutrecht/jwt-angu...](#)

```

1  @RequestMapping(value = "login", method = RequestMethod.POST)
2  public LoginResponse login(@RequestBody LoginRequest loginRequest)
3      throws ServletException {
4      if (loginRequest.name == null || !userDb.exists(loginRequest.name))
5          throw new ServletException("Invalid login request");
6      }
7      return new LoginResponse(Jwts.builder()
8          .claim("roles", userDb.getRoles(loginRequest.name))
9          .signWith(SignatureAlgorithm.HS256, secret)
10         .compact());
11

```

JwtUtil... [origin: lcg0124/cloudodo](#)

```

1  public static String generateToken(String username) {
2      String token = Jwts.builder()
3          .setSubject(username)
4          .claim(CommonConstants.CONTENT_TYPE, "application/json")
5          .claim(CommonConstants.CONTENT_ENCODING, "utf-8")
6          .claim(CommonConstants.RENEWAL, true)
7          .setExpiration(new Date(System.currentTimeMillis() + 1000 * 60 * 60 * 24))
8          .signWith(SignatureAlgorithm.HS256, secret)
9          .compact();
10     return token;
11 }
12

```

... [origin: infiniteautomation/ma...](#)

```

1  public String generateToken(User user) {
2      if (expiry == null) {
3          expiry = new Date(System.currentTimeMillis() + 1000 * 60 * 60 * 24);
4      }
5
6      JwtBuilder builder = this.newTokenBuilder()
7          .claim(USER_ID_CLAIM, user.getId())
8          .claim(USER_TOKEN_VERSION_CLAIM, 1)
9          .signWith(SignatureAlgorithm.HS256, secret);
10     return this.sign(builder);
11 }
12

```

Sets the JWT Claims exp [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

### setSubject

Sets the JWT Claims sub [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

### setIssuedAt

Sets the JWT Claims iat [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

### setClaims

Sets the JWT payload to be a JSON Claims instance populated by the provided Claims

### setIssuer

Sets the JWT Claims iss [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

### setId

Sets the JWT Claims jti [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

### setHeaderParam

Applies the specified name/value pair to the header. If a header does not exist, it will be created.

### setAudience

Sets the JWT Claims aud [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

### compressWith

Compresses the JWT body using the specified CompressionCodec. The default is no compression.

### setNotBefore

Sets the JWT Claims nbf [https://tools.ietf.org/html/draft-ietf-oauth-jwt-04#section-4.1.5]

`compressWith`, `setNotBefore`, `setHeader`,  
`setHeaderParams`, `setPayload`, `addClaims`,  
`serializeToJsonWith`

## Popular in Java

[Parsing JSON documents to java classes using gson](#)

[getContentResolver \(Context\)](#)

[onRequestPermissionsResult \(Fragment\)](#)

[getSupportFragmentManager \(FragmentManager\)](#)

[GridBagLayout \(java.awt\)](#)

The GridBagLayout class is a flexible layout manager that aligns components in a grid.

[Runnable \(java.lang\)](#)

Represents a command that can be executed. Often used to run code in a separate thread.

[TreeSet \(java.util\)](#)

TreeSet is an implementation of SortedSet. All optional operations (comparators, etc.) are inherited from SortedSet.

[ThreadPoolExecutor \(java.util.concurrent\)](#)

An ExecutorService that executes each submitted task using one or more threads.

[BasicDataSource \(org.apache.commons.dbcp\)](#)

## Top 25 Plugins for Webstorm

Feedback



```
2 public String getAssertion(final St
3
4     if (algorithm == null) {
5         throw new IllegalStateException
6     }
7
8     return Jwts.builder().signWith(al
9         .setSubject(deviceId)
10        .claim("ten", tenantId)
11        .setExpiration(Date.from(Inst
12        .compact());
13 }
14
```

... origin: org.eclipse.hono/hono...

```
1 @Override
2 public String getAssertion(final St
3
4     if (algorithm == null) {
5         throw new IllegalStateException
6     }
7
8     return Jwts.builder().signWith(al
9         .setSubject(deviceId)
10        .claim("ten", tenantId)
11        .setExpiration(Date.from(Inst
12        .compact());
13 }
14
```

... origin: infiniteautomation/ma...

```
1 public String generateToken(User us
2     if (expirationDate == null) {
3         int expiryDuration = SystemSett
4         expirationDate = new Date(Syste
5     }
6
7     JwtBuilder builder = this.newToke
8         .claim(USER_ID_CLAIM, user.g
9         .claim(USER_PASSWORD_VERSION
10
11     return this.sign(builder);
12 }
13
```

... origin: org.trellisldp/trelli...

```
1 /**
2  * Build a JWT Token.
3  * @param webid the web ID
4  * @param secret the JWT secret
```

```

9      .signWith(privateKeyFor(secret));
10 }
11

```

J... *origin: jimolonly/AuthServer*

```

1  public static String getToken(String username) {
2      return Jwts.builder()
3          .setSubject(username)
4          .claim("roles", "user")
5          .setIssuedAt(new Date())
6          /*过期时间*/
7          .setExpiration(new Date(System.currentTimeMillis() + 1000 * 60 * 60 * 24))
8          .signWith(SignatureAlgorithm.HS256, secret)
9          .compact();
10 }
11

```

... *origin: org.sonarsource.sonar...*

```

1  String refresh(Claims token, int expiresIn) {
2      checkIsStarted();
3      long now = System.currentTimeMillis();
4      JwtBuilder jwtBuilder = Jwts.builder()
5          .for (Map.Entry<String, Object> entry) {
6              jwtBuilder.claim(entry.getKey(), entry.getValue());
7          }
8      jwtBuilder.setExpiration(new Date(now + expiresIn * 1000));
9      return jwtBuilder.signWith(SIGNATURE_ALGORITHM, secret).compact();
10 }
11
12

```

... *origin: org.openksavi.sponge/...*

```

1  @Override
2  public String createAuthToken(User user) {
3      Long authSessionId = currentAuthSessionId;
4
5      JwtBuilder builder = Jwts.builder()
6          .setSubject(user.getName())
7          .compressWith(CompressionCodecs.gzip())
8      String token = builder.compact();
9
10     authTokenSessions.put(authSessionId, token);
11
12     return token;
13 }
14

```

... *origin: com.ufoscout/coreutil...*

```
5     .claim(PAYLOAD_CLAIM_KEY, json)
6     .setIssuedAt(createdDate)
7     .setExpiration(expirationDate)
8     .signWith(signatureAlgorithm, secret)
9     .compact();
10 }
11
```

... *origin: com.ufoscout.coreutil...*

```
1 @Override
2 public <T> Token generate(final String subject, final Date createdDate, final Date expirationDate, final String secret, final String signatureAlgorithm) {
3     String token = Jwts.builder()
4         .setSubject(subject)
5         .claim(PAYLOAD_CLAIM_KEY, json)
6         .setIssuedAt(createdDate)
7         .setExpiration(expirationDate)
8         .signWith(secret, signatureAlgorithm)
9         .compact();
10     return new Token(token, createdDate);
11 }
12
```

Tok... *origin: sitewhere/sitewhere*

```
1 public String generateToken(IUser user) {
2     try {
3         JwtBuilder builder = Jwts.builder()
4             .setExpiration(getExpirationDate())
5             .claim(CLAIM_GRANTED_AUTHORITY, user.getRole())
6             .return builder.compact();
7     } catch (Throwable t) {
8         throw new SiteWhereException("Unable to generate token", t);
9     }
10 }
11
```

OAu... *origin: trellis-ldp/trellis*

```
1 @Test
2 public void testFilterExpiredJwt() {
3     final Key key = secretKeyFor(SignatureAlgorithm.HS256);
4     final String token = Jwts.builder()
5         .signWith(key).compact();
6     when(mockContext.getHeaderString("Authorization")).thenReturn("Bearer " + token);
7
8     final OAuthFilter filter = new OAuthFilter();
9     assertThrows(NotAuthorizedException.class, () -> filter.doFilter(mockRequest, mockResponse, mockFilterChain));
10 }
11
```



JwtBuilder.claim

[Code Index](#)[Add Tabnine to your IDE \(free\)](#)

### Products

- [Search for Java code](#)
- [Search for JavaScript code](#)

### IDE Plugins

- [IntelliJ IDEA](#)
- [WebStorm](#)
- [Android Studio](#)
- [Eclipse](#)
- [Visual Studio Code](#)
- [PyCharm](#)
- [Sublime Text](#)
- [PhpStorm](#)
- [Vim](#)
- [Atom](#)
- [GoLand](#)
- [RubyMine](#)
- [Emacs](#)
- [Jupyter](#)
- [Rider](#)
- [DataGrip](#)
- [AppCode](#)

### Company

- [About Us](#)
- [Contact Us](#)
- [Careers](#)

### Resources

- [FAQ](#)
- [Blog](#)
- [Tabnine Academy](#)
- [Students](#)
- [Terms of use](#)
- [Privacy policy](#)
- [Java Code Index](#)
- [Javascript Code Index](#)

Get Tabnine for your IDE now



JwtBuilder.claim

Code Index

Add Tabnine to your IDE (free)

