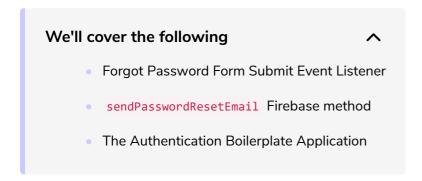
Resetting Passwords

Sometimes users forget their passwords and need a way to reset it. In this lesson I will show you how to pass the users email to Firebase in order to get a password recovery email sent to them.



Forgot Password Form Submit Event Listener

```
// Forgot password form submit event
forgotPasswordForm.addEventListener(`submit`, event => {
    event.preventDefault();
    // Grab value from form
    var emailAddress = document.getElementById(`forgot-password-email`).value;
});

JavaScript
```

sendPasswordResetEmail Firebase method

We invoke the sendPasswordResetEmail Firebase method inside the newly created event listener.

```
// Forgot password form submit event
forgotPasswordForm.addEventListener('submit', event => {
    event.preventDefault()
    // Grab value from form
    var emailAddress = document.getElementById('forgot-password-email').value
    // Send value to Firebase
    firebase.auth().sendPasswordResetEmail(emailAddress)
    .then(() => {
        forgotPasswordForm.reset()
        console.log('Message sent. Please Check your email')
    })
    .catch(error => {
```

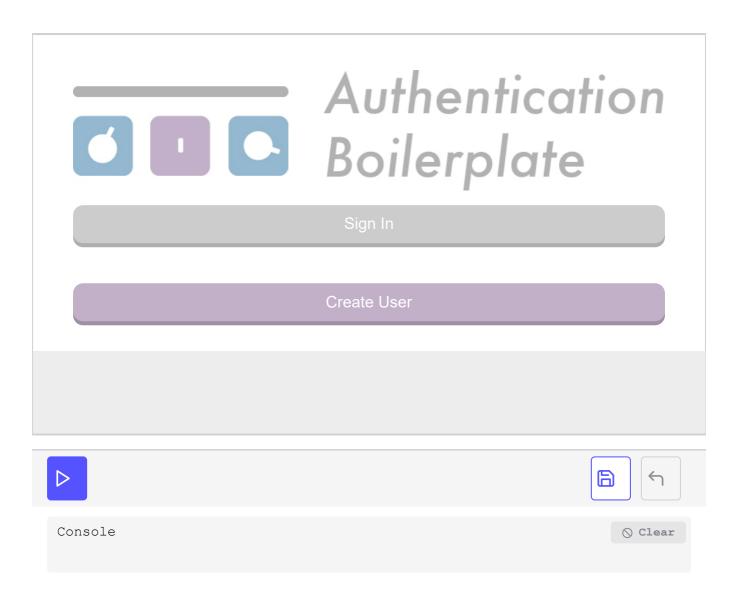
```
13  | console.log(error.message)
14  | })
15  })
```

JavaScript

The Authentication Boilerplate Application

Now our users can reset their passwords. Use the *reset password form* to test this out, but make sure you're resetting a user account that is associated with an email account you have access to.

This code requires the following keys to execute:	
Key:	Value:
apiKey	Not Specified
authDomain	Not Specified
databaseURL	Not Specified
projectId	Not Specified
storageBucket	Not Specified
messagingSenderld	Not Specified
appld	Not Specified
Output	
JavaScript	
HTML	
CSS (SCSS)	



In the next lesson, we cover how to show error and success messages to users. This creates an overall better experience for the users when they are attempting to authenticate themselves within the application.