



Asynchronous Development in ES6

Joe Eames
www.linkedin.com/in/joeeames
[@josepheames](https://twitter.com/josepheames)



pluralsight 
hardcore developer training



Callbacks

Call Stack



Async Process



Callback





Callback Example

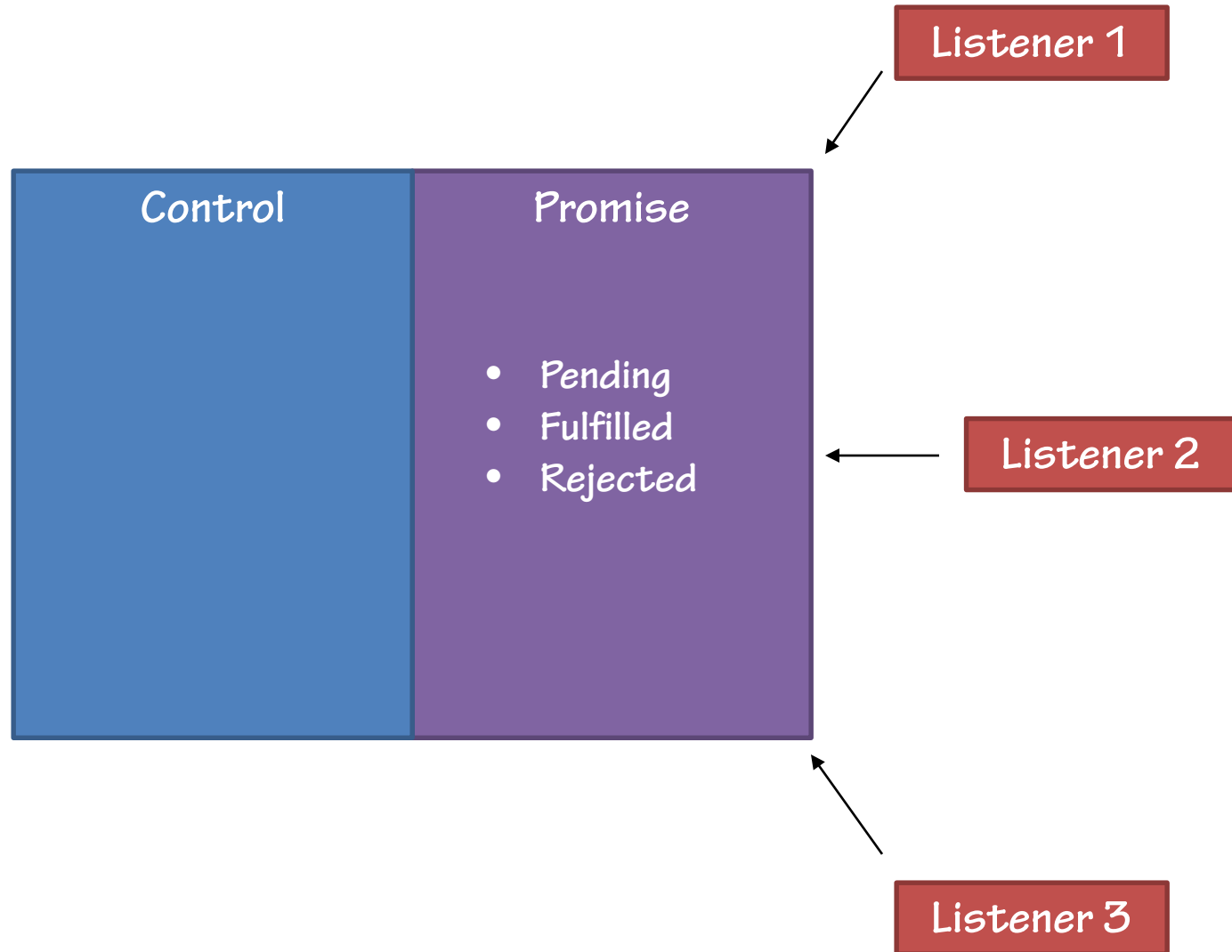
```
function getCompanyFromOrderId(orderId) {  
  getOrder(orderId, function(order) {  
    getUser(order.userId, function(user) {  
      getCompany(user.companyId, function(company) {  
        // do something with company  
      })  
    })  
  })  
}
```




```
function getCompanyFromOrderId(orderId) {  
  try {  
    getOrder(orderId, function(order) {  
      try {  
        getUser(order.userId, function(user) {  
          try {  
            getCompany(user.companyId, function(company) {  
              try {  
                // do something with company  
              } catch(ex) {  
                // handle exception  
              }  
            }  
          } catch(ex) {  
            // handle exception  
          }  
        });  
      } catch(ex) {  
        // handle exception  
      }  
    });  
  } catch(ex) {  
    // handle exception  
  }  
}
```



Promise





```
function getCompanyFromOrderId(orderId) {
  try {
    getOrder(orderId, function(order) {
      try {
        getUser(order.userId, function(user) {
          try {
            getCompany(user.companyId, function(company) {
              try {
                // do something with company
              } catch(ex) {
                // handle exception
              }
            }
          } catch(ex) {
            // handle exception
          }
        });
      } catch(ex) {
        // handle exception
      }
    });
  } catch(ex) {
    // handle exception
  }
}
```



```
function getCompanyFromOrderId(orderId) {  
  getOrder(orderId).then(function(order) {  
    return getUser(order.userId);  
  }).then(function(user) {  
    return getCompany(user.companyId);  
  }).then(function(company) {  
    // do something with company  
  }).then(undefined, function(error) {  
    // handle error  
  })  
}
```



Summary



Promises

A purple circle with a dark purple outline, containing the word "Promises" in white text.

Asynchronous
Generators

A green circle with a dark green outline, containing the text "Asynchronous" and "Generators" in white text, stacked vertically.

Generators
&
Promises

A blue circle with a dark blue outline, containing the text "Generators", "&", and "Promises" in white text, stacked vertically.