0002- Product JS

```
    if ($("#product-details").length > 0) {
      console.log(window.location);
      console.log("Search Params = " + new URLSearchParams(window.location))
      let productId = new URLSearchParams(window.location.search).get('Id');
      console.log(productId);
      renderProductDetails(productId);
    }
```

```


function renderProductDetails(productId) {
  console.log("renderProductDetails");
  console.log(productId);
  EcommerceStore.deployed().then(function(i) {
    i.getProduct(productId).then(function(p) {
      console.log("getProduct");
      console.log(p[4]);
      let content = "";
      ipfs.cat(p[4]).then(function(stream) {

        console.log(stream);
        let content = Utf8ArrayToStr(stream);
        $("#product-desc").append("<div>" + content + "</div>");

      });

      $("#product-image").append("<img src='https://ipfs.io/ipfs/" + p[3] + "' width='250px' />");
      $("#product-price").html(displayPrice(p[7]));
      $("#product-name").html(p[1].name);
      $("#product-auction-end").html(displayEndHours(p[6]));
      $("#product-id").val(p[0]);
      $("#revealing, #bidding").hide();
      let currentTime = getCurrentTimeInSeconds();
      if (currentTime < p[6]) {
        $("#bidding").show();
      } else if (currentTime - (60) < p[6]) {
        $("#revealing").show();
      }
    })
  })
}
```

```
function Utf8ArrayToStr(array) {
    var out, i, len, c;
    var char2, char3;

    out = "";
    len = array.length;
    i = 0;
    while(i < len) {
    c = array[i++];
    switch(c >> 4)
      {
        case 0: case 1: case 2: case 3: case 4: case 5: case 6: case 7:
          // 0xxxxxxx
          out += String.fromCharCode(c);
          break;
        case 12: case 13:
          // 110x xxxx   10xx xxxx
          char2 = array[i++];
          out += String.fromCharCode(((c & 0x1F) << 6) | (char2 & 0x3F));
          break;
        case 14:
          // 1110 xxxx  10xx xxxx  10xx xxxx
          char2 = array[i++];
          char3 = array[i++];
          out += String.fromCharCode(((c & 0x0F) << 12) |
                         ((char2 & 0x3F) << 6) |
                         ((char3 & 0x3F) << 0));
          break;
        default:
          break;
      }
    }

    return out;
}



function getCurrentTimeInSeconds(){
 return Math.round(new Date() / 1000);
}

function displayPrice(amt) {
 return 'Ξ' + web3.fromWei(amt, 'ether');
}
```

```
function displayEndHours(seconds) {
 let current_time = getCurrentTimeInSeconds()
 let remaining_seconds = seconds - current_time;

 if (remaining_seconds <= 0) {
  return "Auction has ended";
 }

 let days = Math.trunc(remaining_seconds / (24*60*60));

 remaining_seconds -= days*24*60*60
 let hours = Math.trunc(remaining_seconds / (60*60));

 remaining_seconds -= hours*60*60

 let minutes = Math.trunc(remaining_seconds / 60);

 if (days > 0) {
  return "Auction ends in " + days + " days, " + hours + ", hours, " + minutes + " minutes";
 } else if (hours > 0) {
  return "Auction ends in " + hours + " hours, " + minutes + " minutes ";
 } else if (minutes > 0) {
  return "Auction ends in " + minutes + " minutes ";
 } else {
  return "Auction ends in " + remaining_seconds + " seconds";
 }
}
```

```
function buildProduct(product, id) {
  console.log("buildProduct");
  console.log(id);
  let node = $("<div/>");
  node.addClass("col-sm-3 text-center col-margin-bottom-1");
  node.append("<img src='http://localhost:8080/ipfs/" + product[3] + "' width='150px' />");
  node.append("<div>" + product[1] + "</div>");
  node.append("<div>" + product[2] + "</div>");
  node.append("<div>" + new Date(product[5] * 1000) + "</div>");
  node.append("<div>" + new Date(product[6] * 1000) + "</div>");
  node.append("<div>Ether " + product[7] + "</div>");
  node.append("<a href='product.html?Id=" + id + "'class='btn btn-primary'>Show</a>")
  return node;
};
```