

KIOSK Printing

Introduction

No need for prerequisites for printing just JavaScript code but the shadow printing is the matter of discussion as it needs some browser configurations on site after deployment

So, at the first shows the steps for how to print and after that we configure the shadow printing.

HTML Code

In receipt design is divided in 3 parts and footer

```
<div id="receiptPrint" style="visibility:hidden">
  <div id="invoice-POS">
    <div id="top">...</div><!--End InvoiceTop-->
    <hr />
    <div id="mid">...</div><!--End Invoice Mid-->
    <hr />
    <div id="bot">...</div><!--End InvoiceBot-->
  </div><!--End Invoice-->
</div>
```

```
<div id="top">
  <div class="logo"></div>
  <div class="info">
    <h2 style="text-align:center;">Top Saloon</h2>
  </div><!--End Info-->
</div><!--End InvoiceTop-->
<hr />
```

```
<div id="mid">
  <div class="info">
    <h2 style="text-align:center;">Receipt</h2>
    <p style="text-align:center;">
      Name : {{customerNameHead}}
    </p>

    <p style="text-align:center;">
      Barber name : {{barberNameHead}}
    </p>

    <p style="text-align:center;">
      Time : {{myDate |date:'medium'}}
    </p>
  </div>
</div><!--End Invoice Mid-->
<hr />
```

In the above 3 photos the top sections that appear 1 time and the structure of the code.

```

<div id="bot">

  <div id="table" style="text-align: center; display: flex; justify-content: center; justify-items: center ">
    <table>
      <tr class="tabletitle">
        <td class="item" style="text-align: left"><h4>Service</h4></td>
        <td class="Hours" style="text-align: right"><h4>Price</h4></td>
      </tr>

      <tbody *ngFor="let m of confirmedServicesToPrint;" style="margin-bottom: 10px;">

        <tr class="service ">
          <td class="text-left" style="text-align: left;"><p class="itemtext"> {{ m.nameEN }}</p></td>
          <td class="text-left" style="text-align: right;"><p class="itemtext"> {{ m.price }} EGP </p></td>
        </tr>
      </tbody>
      <tr class="tabletitle" style="text-align: center;">

        <td class="" style="text-align: left;"><h3>Sub Total</h3></td>
        <td class="" style="text-align: right;"><h4>{{subTotal}} egp</h4></td>
      </tr>

      <tr class="tabletitle" style="text-align: center;">

        <td class="" style="text-align: left;"><h3>discount rate</h3></td>
        <td class="" style="text-align: right;"><h4>{{discountRate}} %</h4></td>
      </tr>

      <tr class="tabletitle" style="text-align: center;">

        <td class="" style="text-align: left;"><h3>Total</h3></td>
        <td class="" style="text-align: right;"><h4>{{totalOrder}} EGP</h4></td>
      </tr>

      <!--added-->
      <tr class="tabletitle" *ngIf="selectedOrderToPrint" style="text-align: center;">

        <td class="" style="text-align: left;"><h3>Service Time taken</h3></td>
        <td class="" style="text-align: right;" *ngIf="selectedOrderToPrint"><h4> {{diffDays}} Mins </h4></td>
      </tr>
    </table>
  </div><!--End Table-->
  <hr />

```

```

<div id="legalcopy" style="text-align: center">
  <p>Thank you for your visit</p>
  <p class="legal">
    Contact us on :
    <strong> 01220040331</strong>
  </p>
</div>

```

The 2 above picture is the part of the iteration shows the total services the system should print it. And the last par is the footer.

All CSS style is custom style there is no need to screenshots of it, but a nice detail to take care of it to (*ngIf) part is ensure that the data in receipt is fully initialized in order to avoid bugging when the UI is rendered in browser before the data is initialized.

TypeScript

There is part before the function of printing which trigger the function of the printing function.

```
finalizeAndPrint() {  
  
    let printContents, popupWin;  
  
    printContents = document.getElementById('receiptPrint').innerHTML;  
    popupWin = window.open('', '_blank', 'top=0,left=0,height=100%,width=auto');  
    popupWin.document.open();  
    popupWin.document.write(`  
        <html>  
            <head>  
                <style>  
  
                </style>  
            </head>  
            <body onload="window.print();window.close()">  
{printContents}</body>  
        </html>`  
  
    );  
    popupWin.document.close();  
  
}
```

The above function is triggered to open the popup window prompt of chrome

- 1- It gets the html section that hold all the data and design to print using id that wrapped up entire section
- 2- And open the popup and pass html file structure as our design printed as HTML file not PDF
- 3- And pass our section content in that in the popup
In this function the popup window appears waiting the order of confirm the print or cancel it.

At the end if we need that popup does not show up this is called shadow printing it mostly used if the you need that print receipt in the host kiosk that client use it to order something in our system and after finish the order the system automatically prints the receipt for what he ordered.

This the configuration for doing it

- 1- Right click on the chrome shortcut
- 2- Select properties
- 3- In target box add "--kiosk-printing" at the end without the quotes
- 4- Those steps do it after close all the chrome tabs from task manger