EXP.NO:4 DATE:	**CREATE FOOD DELIVERY WEBSITE WHERE USERS CAN ORDER FOOD FROM A PARTICULAR RESTAURANT LISTED IN THE WEBSITE**

AIM:

To create food delivery website where users can order food from a particular restaurant listed in the website

ALGORITHM:

1. Create a homepage that displays a list of restaurants available for food delivery.
2. Implement a search bar that allows users to search for specific restaurants by name or location.
3. Upon selecting a restaurant, redirect the user to a page that displays the restaurant's menu.
4. Add functionality for users to select the items they want to order and add them to their cart.
5. Implement a checkout page where users can review their order and enter their delivery information.
6. Add a payment gateway for processing payments securely.
7. Upon successful payment, display a confirmation page with the order details and estimated delivery time.
8. Implement server-side code for storing and retrieving user orders and restaurant menus.
9. Use CSS to style the pages and make them visually appealing.
10. Add JavaScript functionality for handling user interactions, such as updating the cart and displaying order details.
11. Test the web application to ensure it functions properly and is user-friendly.
12. Continuously update and refine the web application as needed, adding new features and improving existing ones such as adding user reviews and ratings, tracking order status, and offering discounts and promotions.

PROGRAM:

Index.html

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Document</title>

  <link rel="stylesheet" href="style.css">

</head>


<body class="">

  <div class="container">

    <header>

      <h1>Tasty Dash</h1>

      <div class="shopping">

        <img src="image/shopping.svg">

        <span class="quantity">0</span>

      </div>

    </header>


    <div class="list">


    </div>

  </div>

  <div class="card">

    <h1>Card</h1>

    <ul class="listCard">

    </ul>
```

```html
    <div class="checkOut">
      <div class="total">0</div>
      <div class="closeShopping">Close</div>
    </div>
  </div>

  <script src="app.js"></script>
</body>
</html>
```

Style.css

```css
body{
  background-color: #E3E7E8;
  font-family: system-ui;
}
.container{
  width: 1000px;
  margin: auto;
  transition: 0.5s;
}
header{
  display: grid;
  grid-template-columns: 1fr 50px;
  margin-top: 50px;
}
header .shopping{
  position: relative;
  text-align: right;
}
header .shopping img{
    width: 40px;
}
header .shopping span{
    background: red;
    border-radius: 50%;
    display: flex;
    justify-content: center;
    align-items: center;
    color: #fff;
    position: absolute;
    top: -5px;
    left: 80%;
    padding: 3px 10px;
}
.list{
    display: grid;
    grid-template-columns: repeat(3, 1fr);
    column-gap: 20px;
    row-gap: 20px;
    margin-top: 50px;
}
.list .item{
    text-align: center;
    background-color: #DCE0E1;
    padding: 20px;
    box-shadow: 0 50px 50px #757676;
    letter-spacing: 1px;
}
.list .item img{
    width: 90%;
    height: 430px;
    object-fit: cover;
}
.list .item .title{
    font-weight: 600;
}
.list .item .price{
    margin: 10px;
}
.list .item button{
    background-color: #1C1F25;
    color: #fff;
    width: 100%;
    padding: 10px;
}
.card{
    position: fixed;
    top:0;
    left: 100%;
    width: 500px;
    background-color: #453E3B;
    height: 100vh;
    transition: 0.5s;
}
.active .card{
    left: calc(100% - 500px);
}
.active .container{
   transform: translateX(-200px);
}
```

```
.card h1{
    color: #E8BC0E;
    font-weight: 100;
    margin: 0;
    padding: 0 20px;
    height: 80px;
    display: flex;
    align-items: center;
}
.card .checkOut{
    position: absolute;
    bottom: 0;
    width: 100%;
    display: grid;
    grid-template-columns: repeat(2, 1fr);

}
.card .checkOut div{
    background-color: #E8BC0E;
    width: 100%;
    height: 70px;
    display: flex;
    justify-content: center;
    align-items: center;
    font-weight: bold;
    cursor: pointer;
}
.card .checkOut div:nth-child(2){
    background-color: #1C1F25;
    color: #fff;
```

```
}
.listCard li{
   display: grid;
   grid-template-columns: 100px repeat(3, 1fr);
   color: #fff;
   row-gap: 10px;
}
.listCard li div{
   display: flex;
   justify-content: center;
   align-items: center;
}
.listCard li img{
   width: 90%;
}
.listCard li button{
   background-color: #fff5;
   border: none;
}
.listCard .count{
   margin: 0 10px;
}
@media only screen and (max-width: 992px) {
   .listProduct{
      grid-template-columns: repeat(3, 1fr);
   }
}


/* mobile */
@media only screen and (max-width: 768px) {
  .listProduct{
    grid-template-columns: repeat(2, 1fr);
  }
}
```

app.js

```
let openShopping = document.querySelector('.shopping');
let closeShopping = document.querySelector('.closeShopping');
let list = document.querySelector('.list');
let listCard = document.querySelector('.listCard');
let body = document.querySelector('body');
let total = document.querySelector('.total');
let quantity = document.querySelector('.quantity');

openShopping.addEventListener('click', ()=>{
  body.classList.add('active');
})
closeShopping.addEventListener('click', ()=>{
  body.classList.remove('active');
})

let products = [
  {
    id: 1,
    name: 'Delicious Trio',
    image: '1.PNG',
  price: 120
},
{
  id: 2,
  name: 'Dunk Duo',
  image: '2.PNG',
  price: 100
},
{
  id: 3,
  name: 'Morning Duo',
  image: '3.PNG',
  price: 200
},
{
  id: 4,
  name: 'South Indian Meals',
  image: '4.PNG',
  price: 230
},
{
  id: 5,
  name: 'Hyderabad Briyani',
  image: '5.PNG',
  price: 320
  },
  {
    id: 6,
    name: 'North Indian Style',
    image: '6.PNG',
    price: 120
  }
];
let listCards  = [];
function initApp(){
  products.forEach((value, key) =>{
    let newDiv = document.createElement('div');
    newDiv.classList.add('item');
    newDiv.innerHTML = `
      <img src="image/${value.image}">
      <div class="title">${value.name}</div>
      <div class="price">Rs.${value.price.toLocaleString()}</div>
      <button onclick="addToCard(${key})">Add To Card</button>`;
    list.appendChild(newDiv);
  })
}
initApp();
function addToCard(key){
  if(listCards[key] == null){
    // copy product form list to list card
    listCards[key] = JSON.parse(JSON.stringify(products[key]));
    listCards[key].quantity = 1;
  }
  reloadCard();
}
function reloadCard(){
  listCard.innerHTML = '';
  let count = 0;
  let totalPrice = 0;
  listCards.forEach((value, key)=>{
    totalPrice = totalPrice + value.price;
    count = count + value.quantity;
    if(value != null){
      let newDiv = document.createElement('li');
      newDiv.innerHTML = `
        <div><img src="image/${value.image}"/></div>
        <div>${value.name}</div>
        <div>Rs.${value.price.toLocaleString()}</div>
        <div>
          <button onclick="changeQuantity(${key}, ${value.quantity - 1})">-
</button>
          <div class="count">${value.quantity}</div>
          <button onclick="changeQuantity(${key}, ${value.quantity +
1})">+</button>
        </div>`;
```

```
            listCard.appendChild(newDiv);
        }
    })
    total.innerText = totalPrice.toLocaleString();
    quantity.innerText = count;
}
function changeQuantity(key, quantity){
    if(quantity == 0){
        delete listCards[key];
    }else{
        listCards[key].quantity = quantity;
        listCards[key].price = quantity * products[key].price;
    }
    reloadCard();
}
```

OUTPUT: