Day 1 Evaluation pdf

Shift 1

Question –

1. You will receive a Json Message of Students in a service Bus queue.
2. Create a Logic app with service Bus trigger which gets the information from service bus queue.
3. Logic app splits the message and send it to another service Bus queue.
4. This other queue message will be picked by a azure function and write code for below business logic:  
   a)  If 10th marks and 12th marks is Greater than 90 % then they will be allocated to their first preferred branch.  
   b) If 10th marks and 12th marks is less than 90% and greater than 80% then they will be allocated to their second preferred branch.  
   c) If 10th marks and 12th marks is less than 80% then they will be allocated to their third preferred branch.
5. Place each student Json in a Storage account Branch folder with file name as Student First Name and Student Last Name and a number at last.

 Data -

[  
  {  
    "firstName": "Aarav",  
    "lastName": "Sharma",  
    "email": "aarav.sharma@example.com",  
    "mobileNumber": "+919812345001",  
    "tenthMarks": 91.2,  
    "twelfthMarks": 88.5,  
    "yearOfPassing": 2024,  
    "branchChoices": ["Computer Science", "Mechanical", "Civil"]  
  },  
  {  
    "firstName": "Neha",  
    "lastName": "Verma",  
    "email": "neha.verma@example.com",  
    "mobileNumber": "+919812345002",  
    "tenthMarks": 89.4,  
    "twelfthMarks": 90.1,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Electronics", "Computer Science", "IT"]  
  },  
  {  
    "firstName": "Rohan",  
    "lastName": "Gupta",  
    "email": "rohan.gupta@example.com",  
    "mobileNumber": "+919812345003",  
    "tenthMarks": 85.5,  
    "twelfthMarks": 87.0,  
    "yearOfPassing": 2024,  
    "branchChoices": ["Mechanical", "Civil", "Automobile"]  
  },  
  {  
    "firstName": "Isha",  
    "lastName": "Singh",  
    "email": "isha.singh@example.com",  
    "mobileNumber": "+919812345004",  
    "tenthMarks": 93.0,  
    "twelfthMarks": 95.4,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Computer Science", "Electronics", "Biomedical"]  
  },  
  {  
    "firstName": "Karan",  
    "lastName": "Mehta",  
    "email": "karan.mehta@example.com",  
    "mobileNumber": "+919812345005",  
    "tenthMarks": 76.3,  
    "twelfthMarks": 81.2,  
    "yearOfPassing": 2024,  
    "branchChoices": ["IT", "Civil", "Mechanical"]  
  },  
  {  
    "firstName": "Priya",  
    "lastName": "Reddy",  
    "email": "priya.reddy@example.com",  
    "mobileNumber": "+919812345006",  
    "tenthMarks": 90.5,  
    "twelfthMarks": 92.3,  
    "yearOfPassing": 2022,  
    "branchChoices": ["Computer Science", "IT", "AI & ML"]  
  },  
  {  
    "firstName": "Aditya",  
    "lastName": "Kumar",  
    "email": "aditya.kumar@example.com",  
    "mobileNumber": "+919812345007",  
    "tenthMarks": 88.9,  
    "twelfthMarks": 86.1,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Electronics", "Telecommunication", "Robotics"]  
  },  
  {  
    "firstName": "Simran",  
    "lastName": "Patel",  
    "email": "simran.patel@example.com",  
    "mobileNumber": "+919812345008",  
    "tenthMarks": 82.7,  
    "twelfthMarks": 85.4,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Biomedical", "Biotech", "Chemical"]  
  },  
  {  
    "firstName": "Yash",  
    "lastName": "Joshi",  
    "email": "yash.joshi@example.com",  
    "mobileNumber": "+919812345009",  
    "tenthMarks": 78.6,  
    "twelfthMarks": 79.8,  
    "yearOfPassing": 2024,  
    "branchChoices": ["Mechanical", "Civil", "Mining"]  
  },  
  {  
    "firstName": "Ananya",  
    "lastName": "Desai",  
    "email": "ananya.desai@example.com",  
    "mobileNumber": "+919812345010",  
    "tenthMarks": 95.0,  
    "twelfthMarks": 94.6,  
    "yearOfPassing": 2022,  
    "branchChoices": ["Computer Science", "AI & ML", "Data Science"]  
  },  
  {  
    "firstName": "Rahul",  
    "lastName": "Chopra",  
    "email": "rahul.chopra@example.com",  
    "mobileNumber": "+919812345011",  
    "tenthMarks": 84.5,  
    "twelfthMarks": 87.5,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Electronics", "IT", "Computer Science"]  
  },  
  {  
    "firstName": "Sneha",  
    "lastName": "Bose",  
    "email": "sneha.bose@example.com",  
    "mobileNumber": "+919812345012",  
    "tenthMarks": 91.8,  
    "twelfthMarks": 93.1,  
    "yearOfPassing": 2024,  
    "branchChoices": ["AI & ML", "Computer Science", "Data Science"]  
  },  
  {  
    "firstName": "Manav",  
    "lastName": "Kapoor",  
    "email": "manav.kapoor@example.com",  
    "mobileNumber": "+919812345013",  
    "tenthMarks": 77.5,  
    "twelfthMarks": 80.0,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Civil", "Mechanical", "Architecture"]  
  },  
  {  
    "firstName": "Tanya",  
    "lastName": "Dube",  
    "email": "tanya.dube@example.com",  
    "mobileNumber": "+919812345014",  
    "tenthMarks": 86.3,  
    "twelfthMarks": 89.7,  
    "yearOfPassing": 2022,  
    "branchChoices": ["Biotech", "Biomedical", "Chemical"]  
  },  
  {  
    "firstName": "Arjun",  
    "lastName": "Nair",  
    "email": "arjun.nair@example.com",  
    "mobileNumber": "+919812345015",  
    "tenthMarks": 88.0,  
    "twelfthMarks": 91.4,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Robotics", "Electronics", "Computer Science"]  
  },  
  {  
    "firstName": "Ritika",  
    "lastName": "Malik",  
    "email": "ritika.malik@example.com",  
    "mobileNumber": "+919812345016",  
    "tenthMarks": 93.4,  
    "twelfthMarks": 94.2,  
    "yearOfPassing": 2024,  
    "branchChoices": ["AI & ML", "Data Science", "Computer Science"]  
  },  
  {  
    "firstName": "Dev",  
    "lastName": "Shetty",  
    "email": "dev.shetty@example.com",  
    "mobileNumber": "+919812345017",  
    "tenthMarks": 85.6,  
    "twelfthMarks": 87.2,  
    "yearOfPassing": 2022,  
    "branchChoices": ["Mechanical", "Automobile", "Civil"]  
  },  
  {  
    "firstName": "Ira",  
    "lastName": "Khan",  
    "email": "ira.khan@example.com",  
    "mobileNumber": "+919812345018",  
    "tenthMarks": 90.1,  
    "twelfthMarks": 92.7,  
    "yearOfPassing": 2024,  
    "branchChoices": ["Computer Science", "IT", "AI & ML"]  
  },  
  {  
    "firstName": "Kabir",  
    "lastName": "Saxena",  
    "email": "kabir.saxena@example.com",  
    "mobileNumber": "+919812345019",  
    "tenthMarks": 81.2,  
    "twelfthMarks": 83.4,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Civil", "Mining", "Mechanical"]  
  },  
  {  
    "firstName": "Diya",  
    "lastName": "Pandey",  
    "email": "diya.pandey@example.com",  
    "mobileNumber": "+919812345020",  
    "tenthMarks": 87.9,  
    "twelfthMarks": 90.0,  
    "yearOfPassing": 2023,  
    "branchChoices": ["Data Science", "Computer Science", "AI & ML"]  
  }  
]

 Function App code –

using System;

using System.Text.Json;

using System.Threading.Tasks;

using Azure.Messaging.ServiceBus;

using Microsoft.Azure.Functions.Worker;

using Microsoft.Extensions.Logging;

using Microsoft.Identity.Client;

using Azure.Storage.Blobs;

using System.Text;

namespace Company.Function;

public class GetDataFromQueue

{

    private readonly ILogger<GetDataFromQueue> \_logger;

    public GetDataFromQueue(ILogger<GetDataFromQueue> logger)

    {

        \_logger = logger;

    }

    [Function(nameof(GetDataFromQueue))]

    public async Task Run(

        [ServiceBusTrigger("target", Connection = "sbnspractice\_SERVICEBUS")]

        ServiceBusReceivedMessage message,

        ServiceBusMessageActions messageActions)

    {

        try

        {

            \_logger.LogInformation("Message Body: {body}", message.Body);

            var studentsData = JsonSerializer.Deserialize<Student>(message.Body);

            string branch = "";

            int marks = (int)(studentsData.tenthMarks + studentsData.twelfthMarks);

            if (marks > 0.90 \* 200)

            {

                branch = studentsData.branchChoices[0];

            }

            else if ((marks < 0.90 \* 200) && (marks > 0.80 \* 200))

            {

                branch = studentsData.branchChoices[1];

            }

            else if (marks < 0.80 \* 200)

            {

                branch = studentsData.branchChoices[2];

            }

            string last4digits = studentsData.mobileNumber.ToString().Substring(studentsData.mobileNumber.Length - 3);

            string filename = string.Concat(studentsData.firstName, studentsData.lastName, last4digits);

            string blobConnectionString = Environment.GetEnvironmentVariable("AzureWebJobsStorage");

            string containerName = "";

            switch (branch)

            {

                case "Computer Science":

                    containerName = "computerscience";

                    break;

                case "Mechanical":

                    containerName = "mechanical";

                    break;

                case "Civil":

                    containerName = "civil";

                    break;

                case "Electronics":

                    containerName = "electronics";

                    break;

                case "IT":

                    containerName = "informationtechnology";

                    break;

                case "Automobile":

                    containerName = "automobile";

                    break;

                case "Biomedical":

                    containerName = "biomedical";

                    break;

                case "AI & ML":

                    containerName = "aiml";

                    break;

                case "Telecommunication":

                    containerName = "telecommunication";

                    break;

                case "Robotics":

                    containerName = "robotics";

                    break;

                case "Chemical":

                    containerName = "chemical";

                    break;

                case "Mining":

                    containerName = "mining";

                    break;

                default:

                    \_logger.LogInformation("The Branch is not valid");

                    break;

            }

            var blobClient = new BlobContainerClient(blobConnectionString, containerName);

            await blobClient.CreateIfNotExistsAsync();

            var blobPath = $"{containerName}/{filename}";

            var blockBlobClient = blobClient.GetBlobClient(blobPath);

            string output = JsonSerializer.Serialize(studentsData);

            using var stream = new MemoryStream(Encoding.UTF8.GetBytes(output));

            await blockBlobClient.UploadAsync(stream, overwrite: true);

            await messageActions.CompleteMessageAsync(message);

        }

        catch (Exception e)

        {

            \_logger.LogInformation($"Exception:{e.Message}");

        }

    }

    public class Student

    {

        public string firstName { get; set; }

        public string lastName { get; set; }

        public string email { get; set; }

        public string mobileNumber { get; set; }

        public double tenthMarks { get; set; }

        public double twelfthMarks { get; set; }

        public double yearOfPassing { get; set; }

        public List<string> branchChoices { get; set; }

    }

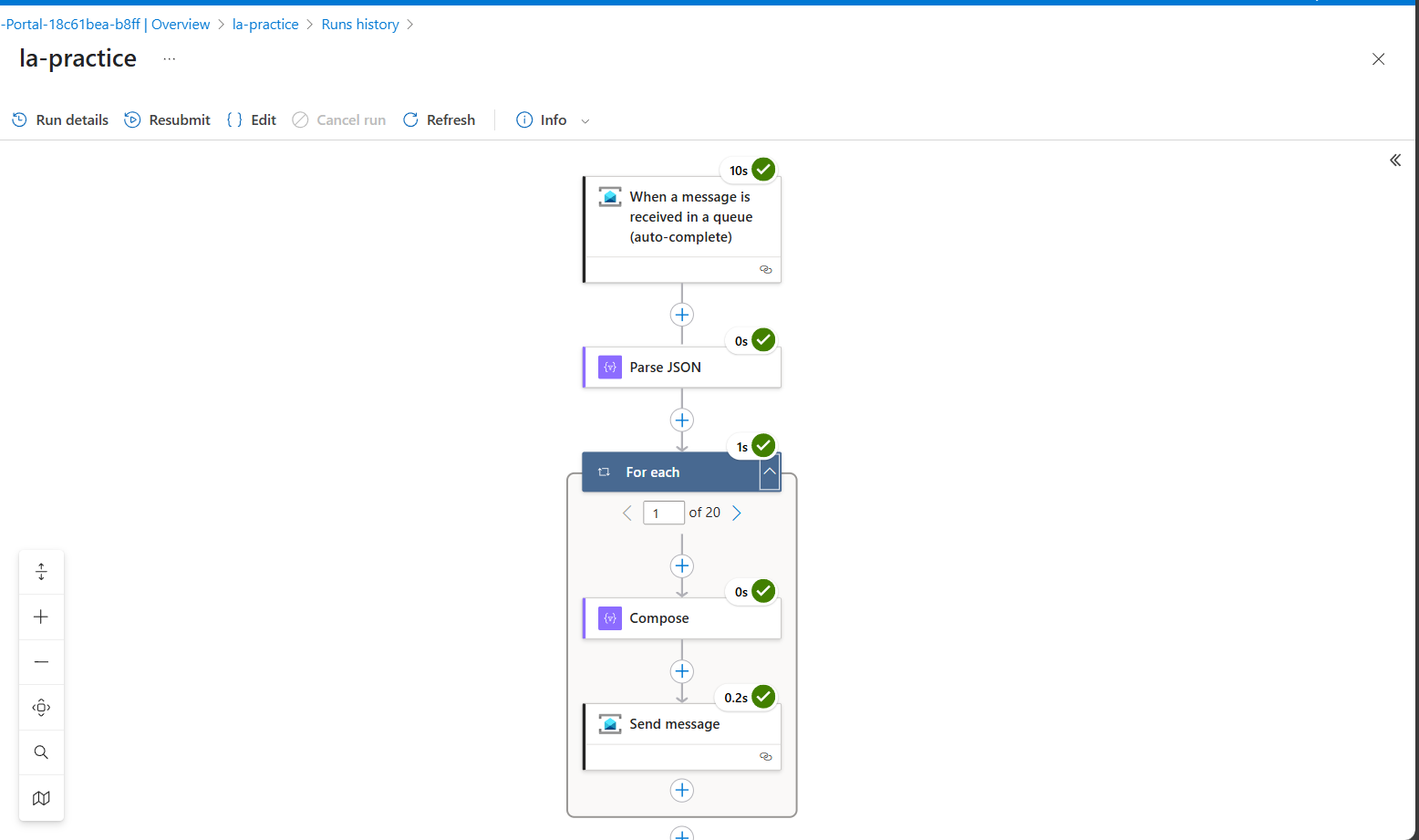
}

Flow chart –

A diagram of an orange box

AI-generated content may be incorrect.

Output Screenshots-



A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Shift 2

Logic App 1

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Logic App 2

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A computer screen shot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.