Booked From	Boarding At	To
MYSURU JN (MYS)	MYSURU JN (MYS)	MGR CHENNAI CTL (MAS)
Start Date* 17-Jul-2025	Departure* 21:00 17-Jul-2025	Arrival* 06:45 18-Jul-2025

PNR	Train No./Name	Class
4221944767	**16022 / KAVERI EXPRESS**	**SLEEPER CLASS (SL)**
Quota	**Distance**	**Booking Date**
GENERAL (GN)	500 KM	10-Jul-2025 22:05:54 HRS

Passenger Details

#	Name	Age	Gender	Booking Status	Current Status
1.	YUGAL	19	M	WL/24	WL/5
2.	AYUSH	20	M	WL/25	WL/6
3.	ABHISHEK	20	M	WL/26	WL/7
4.	SHOBITH	20	M	WL/27	WL/8
5.	LOHITH	19	M	WL/28	WL/9
6.	NISCHAL	19	M	WL/29	WL/10

Acronyms:	RLWL: REMOTE LOCATION WAITLIST	PQWL: POOLED QUOTA WAITLIST	RSWL: ROAD-SIDE WAITLIST

Transaction ID: 100005909995367

IR recovers only 57% of cost of travel on an average.

Payment Details

Ticket Fare	₹ 1,800.00
IRCTC Convenience Fee (Incl. of GST)	₹ 11.80
Total Fare (all inclusive)	₹ 1,811.80

PG Charges as applicable (Additional)

- Beware of fraudulent customer care number. For any assistance, use only the IRCTC e-ticketing Customer care number:14646.

IRCTC Convenience Fee is charged per e-ticket irrespective of number of passengers on the ticket.

* The printed Departure and Arrival Times are liable to change. Please Check correct departure, arrival from Railway Station Enquiry or Dial 139 or SMS RAIL to 139.

- This ticket is booked on a personal User ID, its sale/purchase is an offence u/s 143 of the Railways Act,1989.
- Prescribed original ID proof is required while travelling along with SMS/ VRM/ ERS otherwise will be treated as without ticket and penalized as per Railway Rules.

IRCTC Aadhaar Alert!

Starting July 1, 2025, only Aadhaar authenticated users can book Tatkal Train Tickets.
Authenticate now via "My Account" > "Authenticate User" on
IRCTC website and Mobile app.

Indian Railways GST Details:

Invoice Number:	PS25422194476711	Address:	Indian Railways New Delhi

Supplier Information:

SAC Code:	996421	GSTIN:	07AAAGM0289C1ZL

Recipient Information:

GSTIN:	NA		
Name:	NA	Address:	
Taxable Value:	1776		
CGST Rate:	2.5%	CGST Amount:	0.0
SGST/UGST Rate:		SGST/UGST Amount:	
IGST Rate:	5.0%	IGST Amount:	0.0
Total Tax:			

Place of Supply: NA **State Name/Code of Supplier:** Delhi/DL

<u>**INSTRUCTIONS:**</u>

1. Prescribed Original ID proofs are:- Voter Identity Card / Passport / PAN Card / Driving License / Photo ID card issued by Central / State Govt. / Public Sector Undertakings of State / Central Government ,District Administrations , Municipal bodies and Panchayat Administrations which are having serial number / Student Identity Card with photograph issued by recognized School or College for their students / Nationalized Bank Passbook with photograph /Credit Cards issued by Banks with laminated photograph/Unique Identification Card "Aadhaar", m-Aadhaar, e-Aadhaar. /Passenger showing the Aadhaar/Driving Licence from the "Issued Document" section by logging into his/her DigiLocker account considered as valid proof of identity. (Documents uploaded by the user i.e. the document in "Uploaded Document" section will not be considered as a valid proof of identity).

2. PNRs having fully waitlisted status will be dropped and automatic refund of the ticket amount after deducting the applicable CLERKAGE by Railway shall be credited to the account used for payment for booking of the ticket. Passengers having fully waitlisted e-ticket are not allowed to board the train. However, the names of PARTIALLY waitlisted/confirmed and RAC ticket passenger will appear in the chart.

3. A clerkage charge of Rs.60 per passenger plus GST for AC Classes and Rs.60 per passenger for Non AC classes will be deducted if the ticket remains Waitlisted at the time of Cancellation/Charting.

4. Passengers travelling on a fully waitlisted e-ticket will be treated as Ticketless.

5. Obtain certificate from the TTE /Conductor in case of (a) PARTIALLY waitlisted e-ticket when LESS NO. OF PASSENGERS travel, (b)A.C FAILURE, (c)TRAVEL IN LOWER CLASS. This original certificate must be sent to GGM (IT), IRCTC, Internet Ticketing Centre, 2nd Floor, Tower-D, World Trade Centre, Nauroji Nagar, New Delhi- 110029, after filing TDR online within prescribed time for claiming refund.

6. In case, on a party e-ticket or a family e-ticket issued for travel of more than one passenger, some passengers have confirmed reservation and others are on RAC or waiting list, full refund of fare, less clerkage, shall be admissible for confirmed passengers also subject to the condition that the ticket shall be cancelled online or online TDR shall be filed for all the passengers upto thirty minutes before the scheduled departure of the train.

7. In case train is late more than 3 hours, refund is admissible as per railway refund rules only when TDR is filed by the user before the actual departure of the train at boarding station and passenger has not travelled.

8. In case of train cancellation on its entire run, full refund is granted automatically by the system. However, if the train is cancelled partially on its run or diverted and not touching boarding/destination station, passengers are required to file online TDR within 72 hours of scheduled departure of the train from passengers boarding station.

9. Never purchase e-ticket from unauthorized agents or persons using their personal IDs for commercial purposes. Such tickets are liable to be cancelled and forfeited without any refund of money, under section (143) of the Indian Railway Act 1989. List of authorized agents are available on www.irctc.co.in under 'Find NGet Agents' option.

10. For detail, Rules, Refund rules, Terms & Conditions of E-Ticketing services, Travel Insurance facility etc. Please visit www.irctc.co.in

11. While booking this ticket, you have agreed of having read the Health Protocol of Destination State of your travel. You are again advised to clearly read the Health Protocol advisory of destination state before start of your travel and follow them properly.

12. The FIR forms are available with on board ticket checking staff, train guard and train escorting RPF/GRP staff.

13. Variety of meals available in more than 1500 trains. For delivery of meal of your choice on your seat log on to www.ecatering.irctc.co.in or call 1323 Toll Free. For any suggestions/complaints related to Catering services, contact Toll Free No. 1800-111-321 (07.00 hrs to 22.00 hrs)

14. National Consumer Helpline (NCH) Toll Free Number: 1800-11-400 or 14404

15. You can book unreserved ticket from UTS APP or ATVMs (Automatic Ticket Vending Machines) located in Railway Stations.

16. As per RBI guidelines, the refund of Ticket should be given in the same Bank account, which was used for booking. It is necessary that the Bank Account used for booking online ticket should not be closed at least up to 30 days beyond the date of the journey. If accounts are found closed at the time of processing refund, the refund will be regretted by the Bank.

<u>**Customer Care:**</u>

- For e-ticket booking ,cancellation and refund assistance , Please contact us at 14646 / 08044647999 /08035734999 or raise query at https://equery.irctc.co.in

- Just dial 139 from your landline, mobile & CDMA phones for railway enquiries as well as for giving suggestions/filing complaints on Rail Madad.

- For e-catering, to book and get food delivered on your train berth, please contact us at 1323 (24*7 Hrs Toll Free) or log on to www.ecatering.irctc.co.in.

भारतीय रेल — INDIAN RAILWAYS — TRAINS AT A GLANCE-2025 With Effect From 1 January 2025 ₹100

Print ERS Without Advertisements [X]

7. Implement 0/1 Knapsack problem using Dynamic Programming.

```java
import java.util.Scanner;
public class Dknapsack {
        static int n, m, w[], v[][],value[];
    public static int knap(int i,int j)
    {
        if(i==0||j==0)
        {
            v[i][j] = 0;
        }
        else if(j<w[i])
        {
            v[i][j] = knap(i-1,j);
        }
        else
        {
            v[i][j] = Math.max(knap(i-1,j),value[i]+knap(i-1,j-w[i]));
        }
        return v[i][j];
    }

    public static void optimal(int i,int j)
    {
        if(i>=1 || j>=1) {
            if(v[i][j]!=v[i-1][j])
            {
                System.out.println("Item : "+i);
                j = j-w[i];
                optimal(i-1,j);
            }
            else
            {
                optimal(i-1,j);
            }
        }
    }

    public static void main(String[] args) {
        int profit,i;
        Scanner in = new Scanner(System.in);
        System.out.println("Enter the number of items:");
        n = in.nextInt();
        System.out.println("Enter the capacity of the knapsack:");
        m = in.nextInt();
        w=new int[n+1];
        value=new int[n+1];
        v=new int[n+1][m+1];
        System.out.println("\nEnter weights:");
```

```java
        for(i=1; i<=n; i++)
        {
          w[i]=in.nextInt();
        }
        System.out.println("\nEnter profits:");
        for(i=1; i<=n; i++)
        {
          value[i]=in.nextInt();
        }
    profit = knap(n,m);
    System.out.println("Profit: "+profit);
    System.out.println("Items to be added for Optimal Solution:");
    optimal(n,m);
    in.close();
  }
}
```

4. From a given source vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.

```java
import java.util.Scanner;
public class Dijakstras {
        static int a[][];
        static int n;
        public static void main(String args[])
        {
         Scanner in = new Scanner(System.in);
         System.out.println("Enter the number of vertices:");
         n = in.nextInt();
         System.out.println("Enter the cost adjacency matrix");
         a = new int[n][n];
         for (int i=0;i<n;i++)
         {
           for (int j=0;j<n;j++)
           {
             a[i][j] = in.nextInt();
           }
         }
         System.out.println("\nEnter the source vertex");
         int s=in.nextInt();
         Dijkstra(s);
         in.close();
        }

        public static void Dijkstra(int s)
        {
         int visited[] = new int[n];
         int d[] = new int[n];
         int i,u,v;
         for(i=0;i<n;i++)
         {
           visited[i]=0;
           d[i] = a[s][i];
         }
         visited[s]=1;
         d[s]=0;
         i=1;
         while(i<=n-1)
         {
           u = Extract_Min(visited,d);
           visited[u]=1;
           i++;
           for(v=0;v<n;v++)
           {
             if((d[u]+a[u][v]<d[v]) && visited[v]==0)
```

```java
        d[v]= d[u]+a[u][v];
      }
    }
    for(i=0;i<n;i++)
    {
      if(i!=s)
        System.out.println(s+"->"+i+":"+d[i]);
    }
  }

  public static int Extract_Min(int visited[],int d[])
  {
    int i,j=0,min=999;
    for(i=0;i<n;i++)
      if(d[i]<min && visited[i]==0)
      {
        min = d[i];
        j=i;
      }
      return j;
  }
}
```