```
#include <stdio.h>

#include <stdlib.h>

typedef struct

{

 Char *dayName;

 int date;

 Char *activity;

}week;

Void create(week *day)

{

 Day->dayName = (char *)malloc(sizeof(char) * 20);

 Day->activity = (char *)malloc(sizeof(char) * 100);

 Printf(“Enter the day name: “);

 Scanf(“%s”, day->dayName);

 Printf(“Enter the date: “);
```

```
 Scanf("%d", &day->date);

 Printf("Enter the activity for the day: ");

 Scanf(" %[^\n]s", day->activity);

}

Void read(week *calendar, int size)

{ int i;

 For (i = 0; i< size; i++) {

Printf("Enter details for Day %d:\n", i+ 1);

Create(&calendar[i]);

 }
Void display(week *calendar, int size)

{

 Int i;

 Printf("\nWeek's Activity Details:\n");

 Printf("---------------------------------------------------\n");
```

```
 Printf(“Dayno\t\tDayname\t\tDate\t\tActivity\n”);

 Printf(“-------------------------------------------------\n”);

 For (i= 0; i< size; i++)

 {

Printf(“%d\t\t”, i+ 1);

Printf(“| %s |\t\t”, calendar[i].dayName);

Printf(“| %d |\t\t”, calendar[i].date);

Printf(“| %s |\t\t”, calendar[i].activity);

Printf(“\n”);

 }

}

int main() {

 int size;

 Week *calendar;
```

```
 Printf(“Enter the number of days in the week: “);

 Scanf(“%d”, &size);

 Calendar = (week *)malloc(sizeof(week) * size);

 If (calendar == NULL) {

Printf(“Memory allocation failed. Exiting program.\n”);

Return 1;

 }

 Read(calendar, size);

 Display(calendar, size);

 Free(calendar);

 Return 0;

}
}
```