```
SocialNet.java:29: error: ';' expected
            System.out.println("Enter name:")
                                              ^
1 error
```

```
SocialNet.java:67: error: reached end of file while parsing
}
 ^
1 error
```

```
SocialNet.java:48: error: not a statement
        System.out.println;
                          ^
1 error
```

```
SocialNet.java:59: error: unclosed string literal
        System.out.println("Total Friends Count:);
                           ^
1 error
```

Failed to compile

```
SocialNet.java:39: error: not a statement
                    netArr[i][j]  "0";
                                  ^
SocialNet.java:39: error: ';' expected
                    netArr[i][j]  "0";
                                    ^
2 errors
```

```
What is the total number of friends in the network:
4
Enter name:
Bob
Enter name:
Alice
Enter name:
Charlie
Enter name:
David
```

Output:

```
         Bob     Alice   Charlie         David
Bob      0       1       1       1
Alice    1       0       1       1
Charlie  1       1       0       1
David    1       1       1       0

Total Friends Count:
Bob      3
Alice    3
Charlie  3
David    3
```

```
What is the total number of friends in the network:
5
Enter name:
Emily
Enter name:
Frank
Enter name:
Grace
Enter name:
Henry
Enter name:
Isabelle
```

Output:

```
         Emily   Frank   Grace   Henry   Isabelle
Emily    0       1       0       1       1
Frank    1       0       1       1       0
Grace    0       1       0       1       1
Henry    1       1       1       0       1
Isabelle         1       0       1       1       0

Total Friends Count:
Emily    3
Frank    3
Grace    3
Henry    3
Isabelle         3
```

```
What is the total number of friends in the network:
3
Enter name:
Jessica
Enter name:
Kevin
Enter name:
Linda
```

Output:

```
        Jessica Kevin   Linda
Jessica 0       1       1
Kevin   1       0       1
Linda   1       1       0

Total Friends Count:
Jessica 2
Kevin   2
Linda   2
```

```
What is the total number of friends in the network:
2
Enter name:
Michael
Enter name:
Nicole
```

Output:

```
        Michael Nicole
Michael 0       1
Nicole  1       0

Total Friends Count:
Michael 1
Nicole  1
```

```
What is the total number of friends in the network:
1
Enter name:
Olivia
```

Output:

```
        Olivia
Olivia  0

Total Friends Count:
Olivia  0
```

Pseudocode

```
1. Prompt the user to enter the total number of friends in the network.
2. Validate the input: Ensure the entered value is an integer greater than or equal to 1.
3. Store the number of friends in a variable (numFriends).
4. Create a 2D array (netArr) of size numFriends x numFriends to represent the social network.
5. Create an array (names) of size numFriends to store the names of the friends.
6. Iterate over numFriends, prompting the user to enter each friend's name and storing it in the names array.
7. Iterate over numFriends, comparing each friend to every other friend:
    - If the two friends' names are different, calculate the lexicographic difference between their names.
    - If the lexicographic difference is less than or equal to 12, set the corresponding element in the netArr to "1", indicating a friendship.
    - Otherwise, set the corresponding element in the netArr to "0", indicating no friendship.
8. Print the social network chart:
    - Print a header row with the names of the friends.
    - For each friend, print their name followed by their relationships with all other friends.
9. Print the total number of friends each person has:
    - For each friend, calculate their total number of friends by counting the "1"s in their corresponding row of the netArr.
    - Print the friend's name and their total number of friends.
```