Int 1 blueprint/task

1. Admin login function, Boolean
   - look for B!K3U or error message
   - only 3 errors allowed, do while in if else
   - Exit program if 3 errors
   - Return True if password False if fail 3 times

if login2 != True

else if

2. Org. Name function, int
   - Prompt f/ Organization name

3. Ride dist. Function, float
   - Prompt f/ miles
   - only allow 1-200, Give error code
   - re prompt till correct value

4. Ride cost Function, float
   - Enter cost of bike for "inputted orgz"
   - Has to be 50-250$ error, Repeat
   - Display bike race cost input

5. Jersey cost Function, float
   - Prompt Donated to "orgz"
   - 50-150
   - Display answer

Can these function be put into 1 function and developed into an array? Don't

10. V 100

Charity o/o Function, float

- Percentage of bike sales to Orgz. name
- 5-30
- Display result

the Registration Function, int

- Display o/o, org, Option, Prices
- Enter name
- return attendee

get Jersey

- Prompt do you want Jersey f/ [Jersey Cost]
- Jersey Counter
- n/N, skip
- Y/y Prompt f/ Size, Validate Size
- Add Cost to Sales, Price

Display Pass alot of values, Function Void

- Distance: Price Attendees Sales Charity
  (inputted) (inputted) * (counter) = result *(o/o ÷ 100)= result

- Shirts Sod Sales Charity
  (counter)*(inputted cost) * (o/o ÷ 100)=

Total Sales: Sales + Sales

More

task

Get payment

Give receipt

Main

(Price a attendees) + (Shirts sold · Price) = total sales
↓ registration ↓ Shirts

```
int main
{
  if( adminLogin() == 'True' )
  {  char organization[30];        Get orgz. Name
     orgzName() = organization;
     rideDistance(float distance);
     rideCost(float fee);
     JerseyCost(shirtCost);
     CharityPercentage(double percent);
     theRegistration(attendee);
  While(attendee != "quit")  if admin says quit it will
  {                          end for and stop loop
     getJersey(int shirt);
     getPayment(totalPrice);
     giveReciept();
     theRegistration(attendee); or end for re prompt
  }
  If (attendee == 'quit')
  { adminLogin()
     if(adminLogin() != 'True')
     { Go to Paint: }
  }
  Display()
  Else { Puts(exiting); }

return 0;
}
```

Paint:

Puts(set up race)

Login

```
Bool    adminLogin()
{       unsigned int w=0;
        char pass[5];
      Do {
           puts("enter password");
           sranf("%s", pass);
           w++;
      while(pass != password || w<3);

      IF (pass == password)
        {
          return true;
        }
        return False;
}
```

Mares to define

MAXdist

MAXfee const char password[5]="B!k34u

MINfee

MAX price

Min price

MAX percent

MIN percent

name

```
int orgzName()
{   char nam[30];
    Puts("Enter orgzization name:");
    Scanf("%os", name);
    return name[];
}
```

```
float rideDistance()          Distance Function
{
    float distance = 0;

    DO
        Puts("How many miles 1-200");
        Scanf("%f", &distance);
        while(getchar() != '\n');
        if(distance < 1 && distance > MAXdist)
            { puts("Error"); }
    while(distance < 1 && distance > MAXdist);
    return distance;
}

Float rideCost(int organization)    Fee Function
{
    float entranceFee = 0;
    DO
    { Printf("Enter cost for %s (50-250$)", organization);
        Scanf("%f", &entranceFee);
        while(getchar() != '\n');
        if(entranceFee < MINfee && entranceFee > MAXfee)
        { puts("Error"); }
    while(entranceFee < MINfee && entranceFee > MAXfee);
    Printf("The entrence fee is: %.2f", entranceFee);
    return entranceFee;
}
```

```
float jerseyCost(int organization)   Jersey $ function
{
  float jerseyPrice = 0;

  DO {
    Printf("enter jersey cost for %os: race(50-150)", org);
    scanf("%f", &JerseyPrice);
    while(getchar() != '\n');
    IF (JerseyPrice < MINPrice && JerseyPrice > MAXPrice)
      { puts("error"); }
  while(JerseyPrice < MINPrice && JerseyPrice > MAXPrice);
   Printf("jesey will cost %.2f", JerseyPrice);
  return JerseyPrice;
}

Float charityPercent(int organization) Charity %
{
  Float forCharity = 0;
 DO {
    Printf("enter %f %os charity", organization);
    scanf("%f", &forCharity);
    while(getchar() != '\n');
    IF (forCharity < MINpercent && forCharity > MAXpercent)
      { puts(error); }
    while(forCharity < MINpercent && forCharity > MAXpercent
  Printf("the percentage going to charity is %0.2f", forCharity);
  return forCharity;
}
```

```
                unsigned int shirt=0;
Forgot          char input=0;
                char size=0;                    Jersey

int     getJersey(float jerseyPrice)
{       DO{
            printf("want shirt for $%0.2f (Y or N)", jerseyPrice);
            scanf("%c", &input);
            while getchar() != '\n';
            }
        while( input != 'Y' || 'y' || 'N' || 'n');

    if ( input = 'y' || 'Y')
    { DO{
            puts(" Enter size (s,m,l)");
            scanf("%c", size);
            while getchar() != '\n';
            }
        while (size != 'S' || 's' || 'm' || 'M' || 'l' || 'L');
        shirt++;
    }
    return shirt;
}
```

registration function

```
float fee
int theRegistration(float distance, int organization, float forCharity, prig
{
    unsigned int ride=0;
    char attendee[20];

    printf("register f/ %d.2f raised f/ %os", forCharity, orgz);
printf("ride: %od = distance: %0.2f cost: %0.2f, ridett, dist, fee);
 Printf("you can get jersey f/ $%0.2f", JerseyPrice);

Prompt f/ input
   Scanf("%os", &attendee);
   While getchar(c) != '\n'
   return attendee;
}
```

Payment

```
// identifiers
float getPayment(JerseyPrice, shirt, fee, attendee, forCharity)
{   float totalCost=0;
    float creditCard[..] =0;
    totalCost= (shirt * JerseyPrice) + fee;
    Printf("Total cost is %0.2f Enter card #", totalCost);

    Puts("Enter card #");
    while((i<12-1) && ( =getchar())!='\n')
    {   credidCard[i] = c; }
```

will need to use getchar to turn it into String function.

Maximum Use it to validate card input

```
    totalPrice=(forCharity *100);
    Printf("Thanks alot for purchase %0.2f go to charity",
                                        attendee, totalPrice);
    return totalPrice;
}
```

receipt

w/ identifier

```
Void receipt (fee, shirtCost, forCharity)
{  float totalCost = 0;
   Printf("Race: $%0.2f", fee);
   float donation = 0;
   If (shirt cost > 0)
   { printf("shirt: $%0.2f", shirtCost); }

   totalCost = fee + ShirtCost;
   donation = totalCost * (forCharity * 100);
   printf("total Cost: %0.2f", totalCost);
   Printf("donation: %.2f", donation);
}
```

```
Void  race Summary  function, (float for Charity, fee, etc.)
{     unsigned int race = 0;
      puts ("summary display");
      Printf ("summary where %0.2f goes to Charity", forCharity);
    Printf ("Race : %d", race++);
    Printf ("distance: %.2f", distance);
    Printf("price: %0.2f", fee);
    Printf("People: %d", participants);
    Printf("Sales", totalPrice);
      Display rest of results.

}
```