Question 12.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a design of experiments approach would be appropriate.

I have an online clothing shop and I want to improve the effectiveness of my website in converting visitors into customers. There are various elements of the website that could influence user behavior and purchasing decisions, such as the layout, navigation structure, product descriptions, images, checkout process, and promotional offers.

Using a DOE approach, the online clothing shop could systematically vary these elements across different experimental conditions to determine their impact on key metrics like conversion rate, average order value, and customer retention. I could design experiments where I test different variations of the website's homepage layout, product page design, or checkout flow.

By carefully analyzing the results of these experiments, I can identify the optimal combination of website features and design elements that maximize user engagement and drive more conversions. This might involve finding the most effective placement for call-to-action buttons, optimizing product descriptions and images for clarity and appeal, or streamlining the checkout process to reduce friction and abandoned carts.

Overall, a DOE approach in this scenario would allow my shop to make data-driven decisions about their website design and user experience, leading to improved performance metrics and a more satisfying shopping experience for customers.

Question 12.2

To determine the value of 10 different yes/no features to the market value of a house (large yard, solar roof, etc.), a real estate agent plans to survey 50 potential buyers, showing a fictitious house with different combinations of features. To reduce the survey size, the agent wants to show just 16 fictitious houses. Use R's `FrF2` function (in the `FrF2` package) to find a fractional factorial design for this experiment: what set of features should each of the 16 fictitious houses have? Note: the output of `FrF2` is "1" (include) or "-1" (don't include) for each feature.

```
> house.features
<-c("LargeYard","SolarRoof","Appliances","TermiteBond","Patio","Garage","Fence","Pool","Fir
eplace","HardwoodFloors")
```

```
> ffd1 <-FrF2(nruns=16,nfactors=10,factor.names=house.features)

Error in FrF2(nruns = 16, nfactors = 10, factor.names = house.features) :

  could not find function "FrF2"

> library(FrF2)

Loading required package: DoE.base

Loading required package: grid

Loading required package: conf.design

Registered S3 method overwritten by 'DoE.base':

  method           from

  factorize.factor conf.design


Attaching package: ‘DoE.base’


The following objects are masked from ‘package:stats’:


    aov, lm


The following object is masked from ‘package:graphics’:


    plot.design
```

The following object is masked from ‘package:base’:

```
    lengths

> ffd1 <-FrF2(nruns=16,nfactors=10,factor.names=house.features)

> ffd1

   LargeYard SolarRoof Appliances TermiteBond Patio Garage Fence Pool Fireplace

1          1        -1         -1           1    -1     -1     1    1         1

2          1         1          1          -1     1      1     1   -1        -1

3         -1         1         -1          -1    -1      1    -1    1         1

4         -1        -1          1           1     1     -1    -1   -1        -1

5          1        -1          1          -1    -1      1    -1   -1         1

6         -1        -1          1          -1     1     -1    -1    1         1

7          1         1          1           1     1      1     1    1         1

8          1         1         -1           1     1     -1    -1    1        -1

9         -1         1          1          -1    -1     -1     1    1        -1

10        -1         1         -1           1    -1      1    -1   -1        -1

11        -1        -1         -1           1     1      1     1   -1         1

12        -1         1          1           1    -1     -1     1   -1         1

13         1        -1         -1          -1    -1     -1     1   -1        -1

14        -1        -1         -1          -1     1      1     1    1        -1
```

15	1	1	-1	-1	1	-1	-1	-1	1
16	1	-1	1	1	-1	1	-1	1	-1

	HardwoodFloors
1	1
2	-1
3	-1
4	1
5	1
6	-1
7	1
8	-1
9	1
10	1
11	-1
12	-1
13	-1
14	1
15	1
16	-1

class=design, type= FrF2

```
> ffd2 <-FrF2(nruns=16,nfactors=10,factor.names=house.features,default.levels
=c("N","Y"),seed=set.seed(123))

> ffd2

  LargeYard SolarRoof Appliances TermiteBond Patio Garage Fence Pool Fireplace
HardwoodFloors

1     N     Y     Y     Y   N   N   Y   N     Y     N

2     Y     Y     Y     Y   Y   Y   Y   Y     Y     Y

3     N     Y     N     N   N   Y   N   Y     Y     N

4     Y     N     Y     Y   N   Y   N   Y     N     N

5     Y     N     N     Y   N   N   Y   Y     Y     Y

6     Y     N     N     N   N   N   Y   N     N     N

7     Y     N     Y     N   N   Y   N   N     Y     Y

8     N     N     Y     N   Y   N   N   Y     Y     N

9     Y     Y     N     N   Y   N   N   N     Y     Y

10    Y     Y     N     Y   Y   N   N   Y     N     N

11    N     Y     Y     N   N   N   Y   Y     N     Y

12    N     N     N     N   Y   Y   Y   Y     N     Y

13    N     Y     N     Y   N   Y   N   N     N     Y

14    N     N     Y     Y   Y   N   N   N     N     Y

15    Y     Y     Y     N   Y   Y   Y   N     N     N

16    N     N     N     Y   Y   Y   Y   N     Y     N

class=design, type= FrF2
```

```
> summary(ffd2)

Call:

FrF2(nruns = 16, nfactors = 10, factor.names = house.features,

    default.levels = c("N", "Y"), seed = set.seed(123))


Experimental design of type  FrF2

16  runs


Factor settings (scale ends):

  LargeYard SolarRoof Appliances TermiteBond Patio Garage Fence Pool Fireplace
HardwoodFloors

1        N        N        N         N     N     N     N    N       N           N

2        Y        Y        Y         Y     Y     Y     Y    Y       Y           Y


Design generating information:

$legend

 [1] A=LargeYard      B=SolarRoof      C=Appliances     D=TermiteBond    E=Patio
F=Garage         G=Fence          H=Pool           J=Fireplace      K=HardwoodFloors


$generators

[1] E=AB   F=AC   G=BC   H=AD   J=BCD  K=ABCD
```

Alias structure:

$main

[1] A=BE=CF=DH=JK B=AE=CG C=AF=BG D=AH=GJ E=AB=FG F=AC=EG G=BC=DJ=EF=HK H=AD=GK J=AK=DG K=AJ=GH

$fi2

[1] AG=BF=CE=DK=HJ BD=CJ=EH=FK BH=CK=DE=FJ BJ=CD=EK=FH BK=CH=DF=EJ

The design itself:

	LargeYard	SolarRoof	Appliances	TermiteBond	Patio	Garage	Fence	Pool	Fireplace	HardwoodFloors
1	N	Y	Y	Y	N	N	Y	N	Y	N
2	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
3	N	Y	N	N	N	Y	N	Y	Y	N
4	Y	N	Y	Y	N	Y	N	Y	N	N
5	Y	N	N	Y	N	N	Y	Y	Y	Y
6	Y	N	N	N	N	N	Y	N	N	N
7	Y	N	Y	N	N	Y	N	N	Y	Y
8	N	N	Y	N	Y	N	N	Y	Y	N

9	Y	Y	N	N	Y	N	N	N	Y	Y
10	Y	Y	N	Y	Y	N	N	Y	N	N
11	N	Y	Y	N	N	N	Y	Y	N	Y
12	N	N	N	N	Y	Y	Y	Y	N	Y
13	N	Y	N	Y	N	Y	N	N	N	Y
14	N	N	Y	Y	Y	N	N	N	N	Y
15	Y	Y	Y	N	Y	Y	Y	N	N	N
16	N	N	N	Y	Y	Y	Y	N	Y	N

class=design, type= FrF2

Result:

Based on the generated design, here are the feature settings for each of the 16 fictitious houses:

1. **LargeYard: No, SolarRoof: Yes, Appliances: Yes, TermiteBond: Yes, Patio: No, Garage: No, Fence: Yes, Pool: No, Fireplace: Yes, HardwoodFloors: No**
2. **LargeYard: Yes, SolarRoof: Yes, Appliances: Yes, TermiteBond: Yes, Patio: Yes, Garage: Yes, Fence: Yes, Pool: Yes, Fireplace: Yes, HardwoodFloors: Yes**
3. **LargeYard: No, SolarRoof: Yes, Appliances: No, TermiteBond: No, Patio: No, Garage: Yes, Fence: No, Pool: Yes, Fireplace: Yes, HardwoodFloors: No**
4. **LargeYard: Yes, SolarRoof: No, Appliances: Yes, TermiteBond: Yes, Patio: No, Garage: Yes, Fence: No, Pool: Yes, Fireplace: No, HardwoodFloors: No**
5. **LargeYard: Yes, SolarRoof: No, Appliances: No, TermiteBond: Yes, Patio: No, Garage: No, Fence: Yes, Pool: Yes, Fireplace: Yes, HardwoodFloors: Yes**
6. **LargeYard: Yes, SolarRoof: No, Appliances: No, TermiteBond: No, Patio: No, Garage: No, Fence: Yes, Pool: No, Fireplace: No, HardwoodFloors: No**
7. **LargeYard: Yes, SolarRoof: No, Appliances: Yes, TermiteBond: No, Patio: No, Garage: Yes, Fence: No, Pool: No, Fireplace: Yes, HardwoodFloors: Yes**
8. **LargeYard: No, SolarRoof: No, Appliances: Yes, TermiteBond: No, Patio: Yes, Garage: No, Fence: No, Pool: Yes, Fireplace: Yes, HardwoodFloors: No**

9. **LargeYard: Yes, SolarRoof: Yes, Appliances: No, TermiteBond: No, Patio: Yes, Garage: No, Fence: No, Pool: No, Fireplace: Yes, HardwoodFloors: Yes**
10. **LargeYard: Yes, SolarRoof: Yes, Appliances: No, TermiteBond: Yes, Patio: Yes, Garage: No, Fence: No, Pool: Yes, Fireplace: No, HardwoodFloors: No**
11. **LargeYard: No, SolarRoof: Yes, Appliances: Yes, TermiteBond: No, Patio: No, Garage: No, Fence: Yes, Pool: Yes, Fireplace: No, HardwoodFloors: Yes**
12. **LargeYard: No, SolarRoof: No, Appliances: No, TermiteBond: No, Patio: Yes, Garage: Yes, Fence: Yes, Pool: Yes, Fireplace: No, HardwoodFloors: Yes**
13. **LargeYard: No, SolarRoof: Yes, Appliances: No, TermiteBond: Yes, Patio: No, Garage: Yes, Fence: No, Pool: No, Fireplace: No, HardwoodFloors: Yes**
14. **LargeYard: No, SolarRoof: No, Appliances: Yes, TermiteBond: Yes, Patio: Yes, Garage: No, Fence: No, Pool: No, Fireplace: No, HardwoodFloors: Yes**
15. **LargeYard: Yes, SolarRoof: Yes, Appliances: Yes, TermiteBond: No, Patio: Yes, Garage: Yes, Fence: Yes, Pool: No, Fireplace: No, HardwoodFloors: No**
16. **LargeYard: No, SolarRoof: No, Appliances: No, TermiteBond: Yes, Patio: Yes, Garage: Yes, Fence: Yes, Pool: No, Fireplace: Yes, HardwoodFloors: No**

These combinations allow the real estate agent to present a reduced set of 16 fictitious houses to the potential buyers while still capturing the influence of the 10 different features on the perceived market value of a house.

Question 13.1

For each of the following distributions, give an example of data that you would expect to follow this distribution (besides the examples already discussed in class).

a. **Binomial**
b. **Geometric**
c. **Poisson**
d. **Exponential**
e. **Weibull**

a. Binomial Distribution: This distribution represents the number of successes in a fixed number of independent Bernoulli trials, where each trial has the same probability of success.

Example: The number of heads obtained when flipping a fair coin 10 times. Each flip of the coin is a Bernoulli trial with two possible outcomes (heads or tails), and the probability of getting heads is constant for each trial.

b. Geometric Distribution: This distribution represents the number of trials needed to achieve the first success in a sequence of independent Bernoulli trials, each with the same probability of success.

Example: The number of attempts needed to score a goal in a soccer match, where each attempt is an independent trial with the same probability of scoring a goal.

c. Poisson Distribution: This distribution represents the number of events occurring in a fixed interval of time or space, where events happen independently at a constant rate.

Example: The number of customers arriving at a store in a 15-minute interval, where the arrival rate of customers follows a constant average rate per unit of time.

d. Exponential Distribution: This distribution represents the time between events in a process that occurs at a constant rate, where events occur independently of each other.

Example: The time between arrivals of buses at a bus stop during a specific time period, assuming that buses arrive according to a constant average rate.

e. Weibull Distribution: This distribution is often used to model the time until failure of a system or component, where the hazard rate (the probability of failure per unit time) changes over time.

Example: The time until a light bulb fails, where the likelihood of failure decreases over time due to factors such as wear and tear or manufacturing defects.