```
Course: Statistical_Inference
Lesson: Power

- Class: text
  Output: "Power. (Slides for this and other Data Science courses may be found at github https://github.com/DataScienceSpecialization/courses/. If you care to use them, they must be downloaded as a zip file and viewed locally. This lesson corresponds to 06_Statistical_Inference/11_Power.)"

- Class: text
  Output: In this lesson, as the name suggests, we'll discuss POWER, which  is the probability of rejecting the null hypothesis when it is false, which is good and proper.

- Class: text
  Output: Hence you want more POWER.

- Class: text
  Output: Power comes into play when you're designing an experiment, and in particular, if you're trying to determine if a null result (failing to reject a null hypothesis) is meaningful. For instance, you might have to determine if your sample size was big enough to yield a meaningful, rather than random, result.

- Class: text
  Output: Power gives you the opportunity to detect if your ALTERNATIVE hypothesis is true.

- Class: mult_question
  Output: Do you recall the definition of a Type II error?  Remember, errors are bad.
  AnswerChoices: Rejecting a true null hypothesis; Accepting a false null hypothesis; Miscalculating a t score; Misspelling the word hypothesis
  CorrectAnswer: Accepting a false null hypothesis
  AnswerTests: omnitest(correctVal='Accepting a false null hypothesis')
  Hint: Remember the courtroom example? Letting a guilty person walk, accepting the null hypothesis of innocence, is a Type II error.

- Class: text
  Output: Beta is the probability of a Type II error, accepting a false null hypothesis;  the complement of this is obviously (1 - beta) which represents the probability of rejecting a false null hypothesis. This is good and this is POWER!

- Class: text
  Output: Recall our previous example involving the Respiratory Distress Index and sleep disturbances. Our null hypothesis H_0 was that mu = 30 and our alternative hypothesis H_a was that mu > 30.

- Class: mult_question
  Output: Which of the following expressions represents our test statistic under this null hypothesis? Here X' represents the sample mean, s is the sample std deviation, and n is the sample size. Assume X' follows a t distribution.
  AnswerChoices: (X'-30)/(s/sqrt(n)); X'/(s^2/n); (X'-30)/(s^2/n); 30/(s/sqrt(n))
  CorrectAnswer:  (X'-30)/(s/sqrt(n))
  AnswerTests: omnitest(correctVal='(X\'-30)/(s/sqrt(n))')
  Hint:  (X'-30)/(s/sqrt(n)) measures the number of standard errors the sample mean is from the mean hypothesized by H_0.

- Class: mult_question
  Output: In the expression for the test statistic (X'-30)/(s/sqrt(n)) what does (s/sqrt(n)) represent?
  AnswerChoices: a standard error; a standard bearer; a standard variance; a standard sample; a standard measure
  CorrectAnswer:  a standard error
  AnswerTests: omnitest(correctVal='a standard error')
  Hint:  Since (X'-30)/(s/sqrt(n)) measures the number of standard errors the sample mean is from the mean hypothesized by H_0, the denominator is the standard error of the sample mean.

- Class: text
  Output: Suppose we're testing a null hypothesis H_0 with an alpha level of .05. Since
```

```
  H_a proposes that mu > 30 (the mean hypothesized by H_0), power is the probability that the true mean mu is greater than the (1-alpha) quantile or qnorm(.95). For simplicity, assume  we're working with normal distributions of which we know the variances.

- Class: figure
  Output: Here's the picture we've used a lot in these lessons. As you know, the shaded portion represents 5% of the area under the curve. If a test statistic fell in this shaded portion we would reject H_0  because the sample mean is too far from the mean (center) of the distribution hypothesized by H_0. Instead we would favor H_a, that mu > 30. This happens with probability .05.
  Figure: conf_5pct.R
  FigureType: new

- Class: text
  Output: You might well ask, "What does this have to do with POWER?" Good question. We'll look at some pictures to show you.

- Class: text
  Output: First we have to emphasize a key point. The two hypotheses, H_0 and H_a, actually represent two distributions since they're talking about means or centers of distributions. H_0 says that the mean is mu_0 (30 in our example) and H_a says that the mean is mu_a.

- Class: text
  Output: We're assuming normality and equal variance, say sigma^2/n, for both hypotheses, so under H_0, X'~ N(mu_0, sigma^2/n) and under H_a, X'~ N(mu_a, sigma^2/n).

- Class: figure
  Output: Here's a picture with the two distributions. We've drawn a vertical line at our favorite spot, at the 95th percentile of the red distribution. To the right of the line lies 5% of the red distribution.
  Figure: twoDistros.R
  FigureType: new

- Class: mult_question
  Output: Quick quiz! Which distribution represents H_0?
  AnswerChoices: the red; the blue
  CorrectAnswer:  the red
  AnswerTests: omnitest(correctVal='the red')
  Hint:  The two distributions have the same spread (variance). They differ in their means (centers). Which has a mean equal to that hypothesized by H_0?

- Class: mult_question
  Output: Which distribution represents H_a?
  AnswerChoices: the red; the blue
  CorrectAnswer:  the blue
  AnswerTests: omnitest(correctVal='the blue')
  Hint:  The two distributions have the same spread (variance). They differ in their means (centers). Which has a mean different than that hypothesized by H_0?

- Class: mult_question
  Output: From the picture, what is the mean proposed by H_a?
  AnswerChoices: 30; 28; 32; 36;
  CorrectAnswer:  32
  AnswerTests: omnitest(correctVal='32')
  Hint:  At what value of mu does does the center (highest point) of the blue distribution fall?

- Class: text
  Output: See how much of the blue distribution lies to the right of that big vertical line?

- Class: text
  Output: That, my friend, is POWER!

- Class: text
  Output: It's the area under the blue curve (H_a) to the right of the vertical line.
```

```
- Class: figure
  Output: Note that the placement of the vertical line depends on the null distribution. Here's another picture with fatter distributions. The vertical line is still at the 95th percentile of the null (red) distribution and 5% of the distribution still lies to its right. The line is calibrated to mu_0 and the variance.
  Figure: twoDistros_fat.R
  FigureType: new

- Class: figure
  Output: Back to our original picture.
  Figure: twoDistros.R
  FigureType: new

- Class: cmd_question
  Output:  We've shamelessly stolen plotting code from the slides so you can see H_a in action. Let's look at pictures before we delve into numbers. We've fixed mu_0 at 30, sigma (standard deviation) at 4 and n (sample size) at 16. The function myplot just needs an alternative mean, mu_a, as argument. Run myplot now with an argument of 34 to see what it does.
  CorrectAnswer: myplot(34)
  AnswerTests: omnitest(correctExpr='myplot(34)')
  Hint: Type myplot(34) at the command prompt.

- Class: cmd_question
  Output:  The distribution represented by H_a moved to the right, so almost all (100%) of the blue curve is to the right of the vertical line, indicating that with mu_a=34, the test is more powerful, i.e., there's a higher probability that it's correct to reject the null hypothesis since it appears false. Now try myplot with an argument of 33.3.
  CorrectAnswer: myplot(33.3)
  AnswerTests: omnitest(correctExpr='myplot(33.3)')
  Hint: Type myplot(33.3) at the command prompt.

- Class: cmd_question
  Output:  This isn't as powerful as the test with mu_a=34 but it makes a pretty picture. Now try myplot with an argument of 30.
  CorrectAnswer: myplot(30)
  AnswerTests: omnitest(correctExpr='myplot(30)')
  Hint: Type myplot(30) at the command prompt.

- Class: text
  Output: Uh Oh! Did the red curve disappear? No. it's just under the blue curve. The power now, the area under the blue curve to the right of the line, is exactly 5% or alpha!


- Class: text
  Output: So what did we learn?

- Class: text
  Output: First, power is a function that depends on a specific value of an alternative mean, mu_a, which is any value greater than mu_0, the mean hypothesized by H_0. (Recall that H_a specified mu>30.)

- Class: text
  Output: Second, if mu_a is much bigger than mu_0=30 then the power (probability) is bigger than if mu_a is close to 30. As mu_a approaches 30, the mean under H_0, the power approaches alpha.

- Class: cmd_question
  Output:  Just for fun try myplot with   an argument of 28.
  CorrectAnswer: myplot(28)
  AnswerTests: omnitest(correctExpr='myplot(28)')
  Hint: Type myplot(28) at the command prompt.

- Class: text
  Output: We see that the blue curve has moved to the left of the red, so the area under it, to the right of the line, is less than the 5% under the red curve. This then is even less powerful and contradicts H_a so it's not worth looking at.
```

```yaml
- Class: figure
  Output: Here's a picture of the power curves for different sample sizes. Again, this uses code "borrowed" from the slides. The alternative means, the mu_a's, are plotted along the horizontal axis and power along the vertical.
  Figure: powerCurve_n.R
  FigureType: new

- Class: mult_question
  Output: What does the graph show us about mu_a?
  AnswerChoices: as it gets bigger, it gets more powerful; as it gets bigger, it gets less powerful; power is independent of mu_a
  CorrectAnswer: as it gets bigger, it gets more powerful
  AnswerTests: omnitest(correctVal='as it gets bigger, it gets more powerful')
  Hint: As you move to the right along the horizontal axis, does the line go up or down?

- Class: mult_question
  Output: What does the graph show us about sample size?
  AnswerChoices: as it gets bigger, it gets more powerful; as it gets bigger, it gets less powerful; power is independent of sample size
  CorrectAnswer: as it gets bigger, it gets more powerful
  AnswerTests: omnitest(correctVal='as it gets bigger, it gets more powerful')
  Hint: You have to check the color key. The purple line represents a bigger sample size than the red or blue. This purple line goes up faster than any of the other lines.

- Class: text
  Output: Now back to numbers. Our test for determining rejection of H_0 involved comparing a test statistic, namely Z=(X'-30)/(sigma/sqrt(n)), against some quantile, say Z_95, which depended on our level size alpha (.05 in this case). H_a proposed that mu > mu_0, so we tested if Z>Z_95.  This is equivalent to X' > Z_95 * (sigma/sqrt(n)) + 30, right?

- Class: text
  Output: Recall that nifty R function pnorm, which gives us the probability that a value drawn from a normal distribution is greater or less than/equal to a specified quantile argument depending on the flag lower.tail. The function also takes a mean and standard deviation as arguments.

- Class: text
  Output: Suppose we call pnorm with the quantile 30 + Z_95 * (sigma/sqrt(n)) and specify mu_a as our mean argument. This would return a probability which we can interpret as POWER. Why?

- Class: figure
  Output: Recall our picture of two distributions. 30 + Z_95 * (sigma/sqrt(n)) represents the point at which our vertical line falls. It's the point on the null distribution at the (1-alpha) level.
  Figure: twoDistros.R
  FigureType: new

- Class: mult_question
  Output: Study this picture. Calling pnorm with 30 + Z_95 * (sigma/sqrt(n)) as the quantile and mu_a, say 32, as the mean and lower.tail=FALSE does what?
  AnswerChoices: returns the area under the blue curve to the right of the line; returns the area under the blue curve to the left of the line; returns the area under the red curve to the right of the line; returns the area under the red curve to the left of the line
  CorrectAnswer: returns the area under the blue curve to the right of the line
  AnswerTests: omnitest(correctVal='returns the area under the blue curve to the right of the line')
  Hint: The argument lower.tail=FALSE is a big clue. It indicates we should look to the right of a line. The mean argument is the second clue. It specifies which distribution (red or blue) to examine.

- Class: mult_question
  Output: Let's try some examples now. Before we do, what do we know pnorm will return if we specify a quantile less than the mean?
  AnswerChoices: an answer less than .50; an answer dependent on alpha; an answer dependent on beta; an answer greater than 1
```

```
    CorrectAnswer: an answer less than .50
    AnswerTests: omnitest(correctVal='an answer less than .50')
    Hint: There are several red herrings in the choices. First pnorm will NEVER return a value greater than 1 because density functions by definition have areas equal to 1. We haven't specified an alpha or beta either. The function pnorm just needs a quantile, mean, and standard deviation. By default it looks at the lower tail of the distribution. That leaves one choice.

  - Class: cmd_question
    Output: First, define a variable z as qnorm(.95)
    CorrectAnswer: z <- qnorm(.95)
    AnswerTests: expr_creates_var('z'); omnitest(correctExpr='z <- qnorm(.95)')
    Hint: Type z <- qnorm(.95) at the command prompt.

  - Class: cmd_question
    Output: Run pnorm now with the quantile 30+z, mean=30, and lower.tail=FALSE. We've specified sigma and n so that the standard deviation of the sample mean is 1.
    CorrectAnswer: pnorm(30+z,mean=30,lower.tail=FALSE)
    AnswerTests: omnitest(correctExpr='pnorm(30+z,mean=30,lower.tail=FALSE)')
    Hint: Type pnorm(30+z,mean=30,lower.tail=FALSE) at the command prompt.

  - Class: cmd_question
    Output: That's not surprising, is it? With the mean set to mu_0 the two distributions, null and alternative, are the same and power=alpha. Now run pnorm now with the quantile 30+z, mean=32, and lower.tail=FALSE.
    CorrectAnswer: pnorm(30+z,mean=32,lower.tail=FALSE)
    AnswerTests: omnitest(correctExpr='pnorm(30+z,mean=32,lower.tail=FALSE)')
    Hint: Type pnorm(30+z,mean=32,lower.tail=FALSE) at the command prompt.

  - Class: text
    Output: See how this is much more powerful? 64% as opposed to 5%. When the sample mean is quite different from (many standard errors greater than)  the mean hypothesized by the null hypothesis, the probability of rejecting H_0 when it is false is much higher. That is power!

  - Class: figure
    Output: Let's look again at the portly distributions.
    Figure: twoDistros_fat.R
    FigureType: new

  - Class: mult_question
    Output: With this standard deviation=2 (fatter distribution) will power be greater or less than with the standard deviation=1?
    AnswerChoices: greater; less than; the same
    CorrectAnswer:  less than
    AnswerTests: omnitest(correctVal='less than')
    Hint:  A greater standard deviation means more variability in the data so the test will be less powerful.

  - Class: cmd_question
    Output:  To see this, run pnorm now with the quantile 30+z, mean=32 and sd=1. Don't forget to set lower.tail=FALSE so you get the right tail.
    CorrectAnswer: pnorm(30+z,mean=32,sd=1,lower.tail=FALSE)
    AnswerTests: omnitest(correctExpr='pnorm(30+z,mean=32,sd=1,lower.tail=FALSE)')
    Hint: Type pnorm(30+z,mean=32,sd=1,lower.tail=FALSE) at the command prompt.

  - Class: cmd_question
    Output:  Now run pnorm now with the quantile 30+z*2, mean=32 and sd=2. Don't forget to set lower.tail=FALSE so you get the right tail.
    CorrectAnswer: pnorm(30+z*2,mean=32,sd=2,lower.tail=FALSE)
    AnswerTests: omnitest(correctExpr='pnorm(30+z*2,mean=32,sd=2,lower.tail=FALSE)')
    Hint: Type pnorm(30+z*2,mean=32,sd=2,lower.tail=FALSE) at the command prompt.


  - Class: mult_question
    Output: See the power drain from 64% to 26% ? Let's review some basic facts about power. We saw before in our pictures that the power of the test depends on mu_a. When H_a specifies that mu > mu_0, then as mu_a grows and exceeds mu_0 increasingly, what happens to power?
```

```yaml
  AnswerChoices: it increases; it decreases; it doesn't change
  CorrectAnswer: it increases
  AnswerTests: omnitest(correctVal='it increases')
  Hint: Remember the different distributions and running pnorm? As mu_a got bigger so did the pnorm result.

- Class: figure
  Output: Here's another question. Recall our power curves from before.
  Figure: powerCurve_n.R
  FigureType: new

- Class: mult_question
  Output: As the sample size increases, what happens to power?
  AnswerChoices: it increases; it decreases; it doesn't change
  CorrectAnswer: it increases
  AnswerTests: omnitest(correctVal='it increases')
  Hint: Look at the picture!


- Class: figure
  Output: Here's another one. More power curves.
  Figure: powerCurve_sigma.R
  FigureType: new

- Class: mult_question
  Output: As variance increases, what happens to power?
  AnswerChoices: it increases; it decreases; it doesn't change
  CorrectAnswer: it decreases
  AnswerTests: omnitest(correctVal='it decreases')
  Hint: Look at the picture!

- Class: figure
  Output: Here's another one. And even more power curves.
  Figure: powerCurve_alpha.R
  FigureType: new

- Class: mult_question
  Output: As alpha increases, what happens to power?
  AnswerChoices: it increases; it decreases; it doesn't change
  CorrectAnswer: it increases
  AnswerTests: omnitest(correctVal='it increases')
  Hint: Look at the picture!

- Class: text
  Output: If H_a proposed that mu != mu_0 we would calculate the one sided power using alpha / 2 in the direction of mu_a (either less than or greater than mu_0). (This is only approximately right, it excludes the probability of getting a large test statistic in the opposite direction of the truth.

- Class: mult_question
  Output: Since power goes up as alpha gets larger would the power of a one-sided test be greater or less than the power of the associated two sided test?
  AnswerChoices: greater; less than; they're the same
  CorrectAnswer: greater
  AnswerTests: omnitest(correctVal='greater')
  Hint: The quantity alpha is bigger than alpha/2 so it's got more power.

- Class: mult_question
  Output: Finally, if H_a specified that mu < mu_0 could we still do the same kind of power calculations?
  AnswerChoices: Yes; No
  CorrectAnswer: Yes
  AnswerTests: omnitest(correctVal='Yes')
  Hint: We just have to look at the right tail and flip all our reasoning.

- Class: text
  Output: Suppose H_a says that mu > mu_0. Then power = 1 - beta = Prob ( X' > mu_0 + z_(1-alpha) * sigma/sqrt(n)) assuming that X'~ N(mu_a,sigma^2/n). Which quantities do we know in this statement, given the context of the problem? Let's work through this.
```

```
- Class: mult_question
  Output: What does the null hypothesis H_0 tell us that the population mean equals?
  AnswerChoices: mu_0; mu_a; beta; alpha
  CorrectAnswer: mu_0
  AnswerTests: omnitest(correctVal='mu_0')
  Hint: H_0 simply proposes a null mean.

- Class: mult_question
  Output: After the null mean mu_0 is proposed what does the designer of the hypothesis
  test specify in order to reject or fail-to-reject H_0? In other words, what is the
  level size of the test?
  AnswerChoices: mu_0; mu_a; beta; alpha
  CorrectAnswer: alpha
  AnswerTests: omnitest(correctVal='alpha')
  Hint: The level size is indicated by a Greek letter.

- Class: text
  Output: So we know that  the quantities mu_0 and alpha are specified by the test
  designer. In the statement 1 - beta = Prob( X' > mu_0 + z_(1-alpha) * sigma/sqrt(n))
  given mu_a > mu_0, mu_0 and alpha are specified, and X' depends on the data. The
  other four quantities, (beta, sigma, n, and mu_a), are all unknown.

- Class: text
  Output: It should be obvious that specifying any three of these unknowns will allow
  us to solve for the missing fourth. Usually, you only try to solve for power (1-beta)
  or the sample size n.

- Class: text
  Output: An interesting point is that  power doesn't need mu_a, sigma and n
  individually.  Instead only sqrt(n)*(mu_a - mu_0) /sigma is needed. The quantity
  (mu_a - mu_0) / sigma is called the EFFECT SIZE. This is the difference in the means
  in standard deviation units. It is unit free so it can be interpreted in different
  settings.

- Class: text
  Output: We'll work through some examples of this now. However, instead of assuming
  that we're working with  normal distributions let's work with t distributions.
  Remember, they're pretty close to normal with large enough sample sizes.

- Class: text
  Output: Power is still a probability, namely P( (X' - mu_0)/(S /sqrt(n)) >
  t_(1-alpha, n-1) given H_a that mu > mu_a ). Notice we use the t quantile instead of
  the z. Also, since the proposed distribution is not centered at mu_0, we have to use
  the non-central t distribution.

- Class: text
  Output: R comes to the rescue again with the function power.t.test. We can omit one
  of the arguments and the function solves for it. Let's first use it to solve for power.

- Class: text
  Output:  We'll run it three times with the same values for n (16) and alpha (.05) but
  different delta and standard deviation values. We'll  show that if delta (difference
  in means) divided by the standard deviation is the same, the power returned will also
  be the same. In other words, the effect size is constant for all three of our tests.

- Class: cmd_question
  Output:  We'll specify a positive delta; this tells power.t.test that H_a proposes
  that mu > mu_0 and so we'll need a one-sided test. First run  power.t.test(n = 16,
  delta = 2 / 4, sd=1, type = "one.sample",  alt = "one.sided")$power .
  CorrectAnswer: power.t.test(n = 16, delta = 2 / 4, sd=1, type = "one.sample",  alt =
  "one.sided")$power
  AnswerTests: omnitest(correctExpr='power.t.test(n = 16, delta = 2 / 4, sd=1, type =
  "one.sample",  alt = "one.sided")$power')
  Hint: Type power.t.test(n = 16, delta = 2 / 4, sd=1, type = "one.sample",  alt =
  "one.sided")$power at the prompt.

- Class: cmd_question
  Output:  Now  change delta to 2 and sd to 4. Keep everything else the same.
```

```
  CorrectAnswer: power.t.test(n = 16, delta = 2 , sd=4, type = "one.sample",  alt = "one.sided")$power
  AnswerTests: omnitest(correctExpr='power.t.test(n = 16, delta = 2 , sd=4, type = "one.sample",  alt = "one.sided")$power')
  Hint: Type power.t.test(n = 16, delta = 2 , sd=4, type = "one.sample",  alt = "one.sided")$power at the prompt.

- Class: cmd_question
  Output:  Same answer, right? Now  change delta to 100 and sd to 200. Keep everything else the same.
  CorrectAnswer: power.t.test(n = 16, delta = 100 , sd=200, type = "one.sample",  alt = "one.sided")$power
  AnswerTests: omnitest(correctExpr='power.t.test(n = 16, delta = 100 , sd=200, type = "one.sample",  alt = "one.sided")$power')
  Hint: Type power.t.test(n = 16, delta = 100 , sd=200, type = "one.sample",  alt = "one.sided")$power at the prompt.

- Class: text
  Output: So keeping the effect size (the ratio delta/sd) constant preserved the power. Let's try a similar experiment except now we'll specify a power we want and solve for the sample size n.

- Class: cmd_question
  Output:  First run power.t.test(power = .8, delta = 2 / 4, sd=1, type = "one.sample",  alt = "one.sided")$n .
  CorrectAnswer: power.t.test(power = .8, delta = 2 / 4, sd=1, type = "one.sample",  alt = "one.sided")$n
  AnswerTests: omnitest(correctExpr='power.t.test(power = .8, delta = 2 / 4, sd=1, type = "one.sample",  alt = "one.sided")$n')
  Hint: Type power.t.test(power = .8, delta = 2 / 4, sd=1, type = "one.sample",  alt = "one.sided")$n.

- Class: cmd_question
  Output:  Now change delta to  2 and sd to 4. Keep everything else the same.
  CorrectAnswer: power.t.test(power = .8, delta = 2, sd=4, type = "one.sample",  alt = "one.sided")$n
  AnswerTests: omnitest(correctExpr='power.t.test(power = .8, delta = 2, sd=4, type = "one.sample",  alt = "one.sided")$n')
  Hint: Type power.t.test(power = .8, delta = 2, sd=4, type = "one.sample",  alt = "one.sided")$n.

- Class: cmd_question
  Output:  Same answer, right? Now  change delta to 100 and sd to 200. Keep everything else the same.
  CorrectAnswer: power.t.test(power = .8, delta = 100 , sd=200, type = "one.sample",  alt = "one.sided")$n
  AnswerTests: omnitest(correctExpr='power.t.test(power = .8, delta = 100 , sd=200, type = "one.sample",  alt = "one.sided")$n')
  Hint: Type power.t.test(power = .8, delta = 100 , sd=200, type = "one.sample",  alt = "one.sided")$n.

- Class: cmd_question
  Output:  Now use power.t.test to find  delta for a power=.8 and n=26 and sd=1
  CorrectAnswer: power.t.test(power = .8, n=26, sd=1, type = "one.sample",  alt = "one.sided")$delta
  AnswerTests: omnitest(correctExpr='power.t.test(power = .8, n=26, sd=1, type = "one.sample",  alt = "one.sided")$delta')
  Hint: Type power.t.test(power = .8, n=26, sd=1, type = "one.sample",  alt = "one.sided")$delta.

- Class: cmd_question
  Output: Not a surprising result, is it? It told you before that with an effect size of .5 and power .8, you need a sample size a little more than 26. Now run it with n=27.
  CorrectAnswer: power.t.test(power = .8, n=27, sd=1, type = "one.sample",  alt = "one.sided")$delta
  AnswerTests: omnitest(correctExpr='power.t.test(power = .8, n=27, sd=1, type = "one.sample",  alt = "one.sided")$delta')
  Hint: Type power.t.test(power = .8, n=27, sd=1, type = "one.sample",  alt = "one.sided")$delta.
```

```yaml

- Class: mult_question
  Output: What do you think will happen if you doubled sd to 2 and ran the same test?
  AnswerChoices: delta will double; delta will halve; delta won't change
  CorrectAnswer: delta will double
  AnswerTests: omnitest(correctVal='delta will double')
  Hint: Since you're doubling the denominator (sd) you have to double the numerator (delta) in order to keep the effect size constant.

- Class: text
  Output: Now for a quick review. We call this the power.u.test since it comes after the power.t.test. LOL.

- Class: mult_question
  Output: 1. The level of a test is specified by what?
  AnswerChoices:  alpha; beta; gamma; delta; None of the others
  CorrectAnswer: alpha
  AnswerTests: omnitest(correctVal='alpha')
  Hint: The level refers to the probability of rejecting  the null hypothesis when it is true. This is the first Greek letter.

- Class: mult_question
  Output: 2. What is a Type II error?
  AnswerChoices:   rejecting a true hypothesis;  rejecting a false hypothesis; accepting a true hypothesis; accepting a false hypothesis
  CorrectAnswer: accepting a false hypothesis
  AnswerTests: omnitest(correctVal='accepting a false hypothesis')
  Hint: Only two of the choices are errors. Eliminate those and then eliminate the Type I error.

- Class: mult_question
  Output: 3. What is power?
  AnswerChoices: thrilling; alpha; beta; gamma; delta; None of the others
  CorrectAnswer: None of the others
  AnswerTests: omnitest(correctVal='None of the others')
  Hint: Power is 1-beta. Is that in the list?

- Class: mult_question
  Output: 4. You're a perfectionist designing an experiment and you want both alpha and beta to be small. Can they both be 0 for this single test?
  AnswerChoices: Yes; No
  CorrectAnswer: No
  AnswerTests: omnitest(correctVal='No')
  Hint: If alpha=0, then you ALWAYS fail to reject (that is, you accept) H_0; if H_0 is false, beta=1. Similarly if beta=0, you ALWAYS reject H_0; if H_0 is true then alpha=1.

- Class: mult_question
  Output: 5. Suppose H_0 proposes mu = mu_0 and H_a proposes that mu < mu_0. You'll test a series of mu_a with power != alpha. Which of the following is NOT true?
  AnswerChoices: mu_0-mu_a > 0;  mu_a-mu_0 < 0; mu_a-mu_0=0 ; huh?
  CorrectAnswer: mu_a-mu_0=0
  AnswerTests: omnitest(correctVal='mu_a-mu_0=0')
  Hint: As in the case in which H_a proposes that mu > mu_0, when mu_a=mu_0, alpha=power. This was excluded in the problem statement so mu_a=mu_0 cannot be true in this case.

- Class: mult_question
  Output: 6. Suppose H_0 proposes mu = mu_0 and H_a proposes that mu < mu_0. Which of the following is true?
  AnswerChoices: the smaller mu_0-mu_a the more powerful the test; the smaller mu_a-mu_0 the more powerful the test; mu_0=mu_a maximizes the power
  CorrectAnswer: the smaller mu_a-mu_0 the more powerful the test
  AnswerTests: omnitest(correctVal='the smaller mu_a-mu_0 the more powerful the test')
  Hint: Here mu_a < mu_0 and the smaller mu_a-mu_0 is, the easier it is to discriminate between mu_a and mu_0.

- Class: mult_question
  Output: 7. Which expression represents the size effect?
```

```
  AnswerChoices: (mu_a - mu_0) / sigma; (mu_a - mu_0) / n; (mu_a - mu_0) / sqrt(sigma);
  (mu_a - mu_0) / sqrt(n)
  CorrectAnswer: (mu_a - mu_0) / sigma
  AnswerTests: omnitest(correctVal='(mu_a - mu_0) / sigma')
  Hint: The size effect is the distance between the two proposed means in standard
  deviation units.

- Class: mult_question
  Output: 8. True or False? More power is better than less power.
  AnswerChoices: True; False
  CorrectAnswer: True
  AnswerTests: omnitest(correctVal='True')
  Hint: Power is good.

- Class: mult_question
  Output: 9. True or False? A larger beta (call it beta_max) is more powerful than a
  smaller beta.
  AnswerChoices: True; False
  CorrectAnswer: False
  AnswerTests: omnitest(correctVal='False')
  Hint: Recall that beta is the probability of an error (Type II), so you'd like a
  smaller beta.

- Class: mult_question
  Output: 10. True or False? The larger the sample size the less powerful the test.
  AnswerChoices: True; False
  CorrectAnswer: False
  AnswerTests: omnitest(correctVal='False')
  Hint: It's usually better to have more data to work with.

- Class: text
  Output: Congrats! You finished this powerful lesson. We hope you feel emPOWERED.

- Class: mult_question
  Output: "Would you like to receive credit for completing this course on
    Coursera.org?"
  CorrectAnswer: NULL
  AnswerChoices: Yes;No
  AnswerTests: coursera_on_demand()
  Hint: ""
```