

```

import moment from 'moment-timezone';
import sinon from 'sinon';
import React from 'react';

import { shallow } from 'enzyme';

import SignupCountdown from '../SignupCountdown';

describe('SignupOpen component', () => {
  let clock;

  beforeEach(() => {
    clock = sinon.useFakeTimers(moment('2017-05-09T00:00').toDate());
  });

  afterEach(() => {
    clock.restore();
  });

  it('renders countdown with ready message when signups are not yet opened', () => {
    const comp = shallow(
      <SignupCountdown
        onOpen={() => {}}
        openDate="2017-05-10T00:00"
      />
    );
    expect(comp.find('Countdown')).toHaveProp('readyMessage', 'years.signupsAreOpen');
  });

  it('renders countdown with will open message when signups are not yet opened', () => {
    const comp = shallow(
      <SignupCountdown
        onOpen={() => {}}
        openDate="2017-05-10T00:00"
      />
    );
    expect(comp.find('Countdown')).toHaveProp('countdownMessage', 'years.signupsWillOpen');
  });

  it('renders countdown with open date when signups are not yet opened', () => {
    const comp = shallow(
      <SignupCountdown
        onOpen={() => {}}
        openDate="2017-05-10T00:00"
      />
    );
  });
});

```

```
);  
expect(comp.find('Countdown')).toHaveProp('date', '2017-05-10T00□00');  
});
```

```
it('triggers onOpen when signups open and were not yet opened', () => {  
  const onOpen = sinon.spy();  
  const comp = shallow(  
    00" onOpen={onOpen} openDate="2017-05-10T00□00" />  
  );  
  comp.find('Countdown').simulate('finish');  
  expect(onOpen.calledOnce).toBeTruthy();  
});
```

```
it('renders countdown with ready message when signups after signups were opened', () => {  
  const comp = shallow(  
    00" onOpen={() => {}}  
    openDate="2017-05-09T00□00"  
  />  
  );  
  expect(comp.find('Countdown')).toHaveProp('readyMessage', 'years.signupsWereClosed');  
});
```

```
it('renders countdown with will close message when signups after signups were opened', () => {  
  const comp = shallow(  
    00" onOpen={() => {}}  
    openDate="2017-05-09T00□00"  
  />  
  );  
  expect(comp.find('Countdown')).toHaveProp('countdownMessage', 'years.signupsCloseIn');  
});
```

```
it('renders countdown with close date when signups after signups were opened', () => {  
  const comp = shallow(  
    00" onOpen={() => {}}  
    openDate="2017-05-09T00□00"  
  />  
  );  
  expect(comp.find('Countdown')).toHaveProp('date', '2017-05-11T00□00');  
});  
});
```