



भौतिक कंप्यूटिंग

शुरुआत करने के लिए गाइड भाग 2





मैं सर्वव्यापी बनाऊंगा

प्रौद्योगिकी के साथ काम करने के लिए समझने हेतु एवं छात्रों की सहायता करने के लिए प्रारंभ करने वालों के लिए मैं सर्वव्यापी बनाऊंगा गाइड तैयार किया गया है।

इसमें एक प्रौद्योगिकी मंच के बारे में एक चरणबद्ध परिचय शामिल है- जेनुइनो 101।

एक मंच के साथ कैसे काम किया जाए हम आपसे यह समझने, सीखने और फिर बाहर जाकर आगे की खोज करने का आग्रह करते हैं ।



इसके अंदर क्या है?

1

ट्यूटोरिअल्स

शुरुआत करना

एलईडी को नियंत्रित करना

फेयरी लाइट्स

पुश बटन

2

व्यावहारिक व क्रियाशील

एक्सेलेरोमीटर

गायरोस्कोप

3

अभ्यास

ब्लूटूथ वर्क्स

पेडोमीटर

विजुअल प्रोग्रामिंग





ट्यूटोरिअल्स



शुरुआत करना

साँफ्टवेयर को स्थापित करना

जेनुइनो 101 बोर्ड खासकर छात्रों के लिए डिज़ाइन किया गया है। यह इस्तेमाल करने में आसान और सरल है और आर्डुइनो संवेदकों एवं कनेक्टिविटी के साथ जैम पैक होकर आता है जो इसे युवा नवप्रवर्तकों एवं निर्माताओं के लिए संपन्न या उत्तम बनाता है।



नो आईडीई का नवीनतम संस्करण (साँफ्टवेयर पैकेज) को इनस्टॉल करें। आप यहां नवीनतम संस्करण देख सकते हैं:

<https://www.arduino.cc/en/Main/Software>



इंटेल् क्यूरी बोर्ड को इनस्टॉल करें। जेनुइनो 101 बोर्ड के उदाहरणों और फंक्शन्स तक पहुँचने के लिए आपको इंटेल् क्यूरी बोर्ड पुस्तकालयों को इनस्टॉल करने की ज़रूरत होगी। टूल्स मेन्यू पर क्लिक करें उसके बाद

A screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for checkmark, refresh, file, upload, and download. The main text area shows the "Blink" example code, which includes a comment in Hindi and a C++ code block for the setup function.

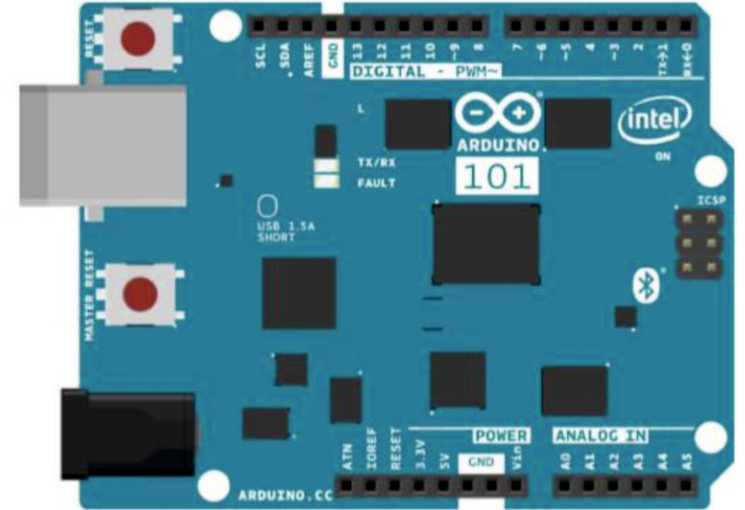
```
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for
 *
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino b
  pinMode(13, OUTPUT);
}
```

बोर्ड को जोड़ना

एक स्टैंडर्ड आर्डुइनो यूएसबी केबल का इस्तेमाल करें। एक आदर्श आर्डुइनो यूएसबी केबल का उपयोग करें। यूएसबी केबल के एक सिरे को बोर्ड पर मौजूद यूएसबी पोर्ट में प्लग करें या लगाएं और यूएसबी केबल के दूसरे सिरे को कंप्यूटर पर मौजूद यूएसबी पोर्ट पर लगाएं।

एक बार जुड़ जाने पर जिस ग्रीन पावर एलईडी पर 'ऑन' लिखा हुआ है वो जलने लगेगा। सुनिश्चित करें की आपने टूल्स-बोर्ड्स- जेनुइनो 101 का चयन किया है और एक सीओएम पोर्ट का चयन किया गया है टूल्स-पोर्ट (अपने जेनुइनो 101 बोर्ड के अनुरूप पोर्ट का चयन करें- यह "सीओएम"(जेनुइनो 101)' जैसा दिखना चाहिए।

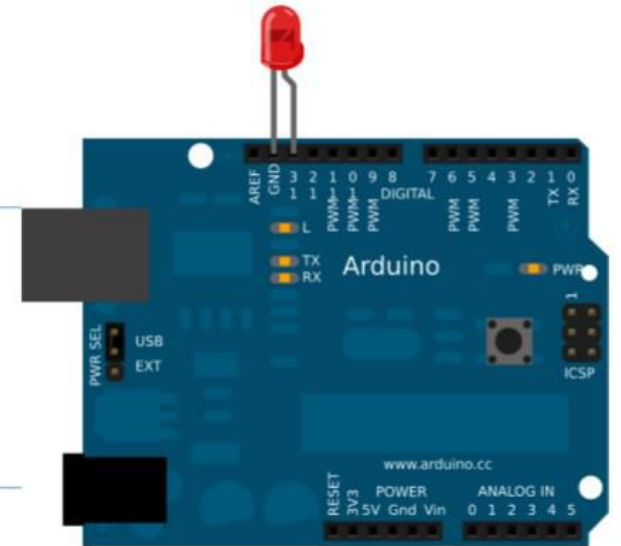


ब्लिंक परीक्षण

आर्डुइनो आईडीई पर जो कोड लिखा जाता है उसे स्केच कहते हैं। अब हमें एक आम स्केच को अपलोड करना होगा ताकि हम यह सुनिश्चित कर सकें की सबकुछ ठीक से काम कर रहा है

1. आईडीई में [फाइल>उधारणों> 01. बेसिक्स> ब्लिंक](#)
2. एक पूर्वस्थापित कोड के साथ [एक नया स्केच खुलेगा।](#)
3. टूलबार पर [अपलोड बटन को क्लिक करें](#)

आप ध्यान दें कि एक छोटी एलईडी आपके बोर्ड पर चमक रही होगी!
अब अपने आप को शाबाशी दें!





एलईडी को नियंत्रित करना

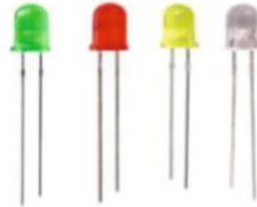


जिन चीज़ों की आपको ज़रूरत पड़ेगी

इस कार्य के लिए आपको जिस चीज़ की भी ज़रूरत होगी यह वो चीज़ें हैं!



2 X जंपर वायres



1 X एलईडी



1 X जेनुइनो बोर्ड



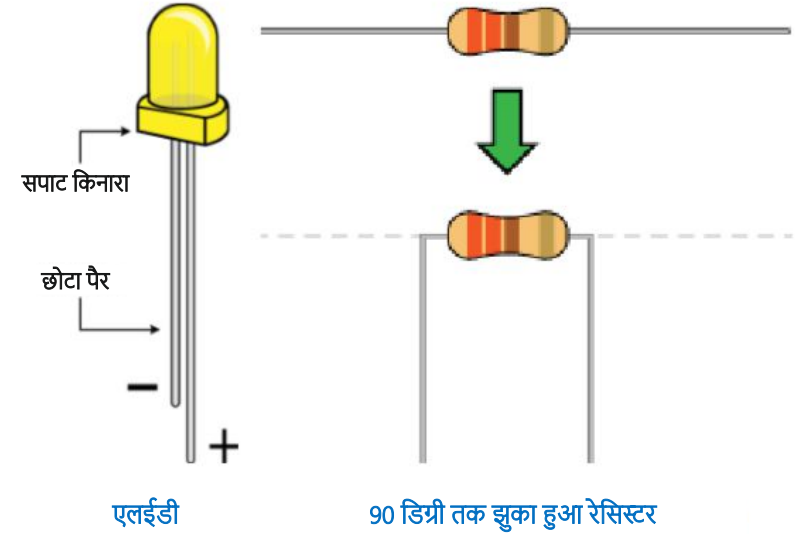
1 X ब्रेडबोर्ड



1 X रेसिस्टर 330 OHM

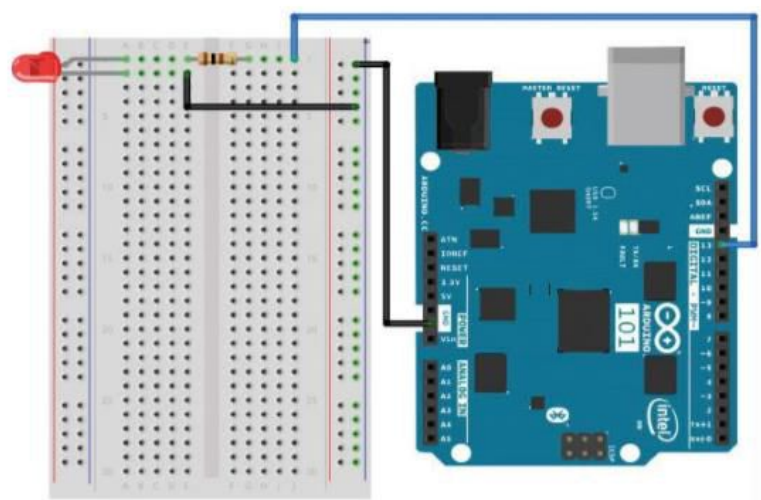
स्ट्रिंग्स लगाना

1. चीज़ों को जोड़ना बहुत आसान है। आपको बस इतना करना है कि घटकों को ब्रेडबोर्ड में घुसाना है।
2. एलईडी घुसाना आसान है, लेकिन, रेसिस्टर को घुसाने के लिए कुछ बदलावों की ज़रूरत होगी।
3. ब्रेडबोर्ड में रेसिस्टर को घुसाने के लिए आपको उसके पैरों को मोड़ना होगा, उसके वास्तविक स्थान से 90 डिग्री एंगल पर।





यह चित्र दिखाता है कि सबकुछ कैसे एक साथ जुड़ा हुआ है



यदि चित्र को समझने में कठिनाई हो तो यह टेबल काम में आ सकता है

घटक	जेनुइनो	ब्रेडबोर्ड	ब्रेडबोर्ड
एलईडी		c2 (+)	c3 (-)
330 रेसिस्टर		a3	(-)
जम्पर वायर	GND	(-)	
जम्पर वायर	PIN 13	e2	

एलईडी एक ध्रुवीकृत घटक है, जिसका अर्थ है कि इसे केवल एक दिशा में सर्किट में जोड़ा जा सकता है।

खैर ... यह सभी वो आवश्यक चीज़ें हैं जिसकी ज़रूरत आपको अपने पहले प्रोजेक्ट के लिए होगी!!





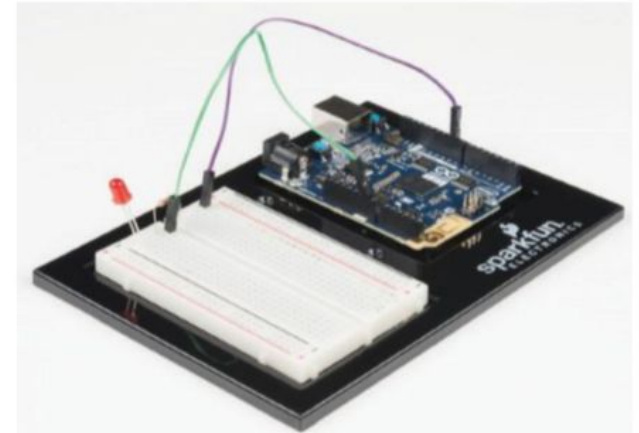
ज़्यादातर आर्डुइनो में एलईडी और रेसिस्टर पिन 13 से पहले से जुड़े होते हैं, इसलिए आपको एक अतिरिक्त सर्किट कि आवश्यकता नहीं पड़ेगी।

लेकिन अगर आप पिन 13 के साथ एक दूसरी एलईडी जोड़ना चाहते हैं या कोई और पिन का इस्तेमाल करना चाहते हैं, तो आप निम्न क़दमों का पालन करें:

- अपने एलईडी के पॉजिटिव साइड (लंबे पाँव) को जेनुइनो डिजिटल पिन 13 से जोड़ें (या अन्य डिजिटल पिन से जोड़ें, मिलान करने के लिए कौड बदलना न भूलें)
- अपने एलईडी के नेगेटिव साइड (छोटे पाँव) को 330 ओम रेसिस्टर (नारंगी-नारंगी-भूरा) से जोड़ें। रेसिस्टर के दूसरे हिस्से को ज़मीन से जोड़ें।

पिन 13 ---- +एलईडी - ----330 ओम __ GND ____

(हम जेनुइनो और एलईडी के बीच में हमेशा रेसिस्टर का इस्तेमाल करते हैं जिससे कि एलईडी को अधिक बिजली के कारण जलने से बचाया जा सके।)



आपका बोर्ड कुछ इस तरह से दिखना चाहिए!

कोड को लिखना

आर्डुइनो आईडीई को खोलें और इस प्रोजेक्ट को कोड करने के लिए तैयार हो जाएँ!

जेनुइनो पिन का इस्तेमाल करने के लिए हमें जेनुइनो को पहले बताना होगा कि जिस पिन का हम इस्तेमाल करना चाह रहे हैं, वो इनपुट होगा या आउटपुट। और ऐसा करने के लिए हम इन-बिल्ट कमांड का इस्तेमाल करेंगे।

पिन मोड ()

अब; पिन नंबर 13 को आउटपुट के रूप में परिभाषित करने के लिए हम निम्न बातें लिखेंगे।

पिन मोड (13, आउटपुट);

पिन को एक आउटपुट के रूप में इस्तेमाल करने के दौरान हमें उसे परिभाषित करने कि आवश्यकता होगी कि आउटपुट अधिक (5 वोल्ट) होगा या कम (0 वोल्ट) होगा। ऐसा करने के लिए हम निम्न कोड लिखेंगे।

डिजिटल राइट (13, उच्च)

अब अगले पृष्ठ पर दिए गए कोड को देखें और इसे अपने जेनुइनो बोर्ड पर अपलोड करें। तब आपको एलईडी ऑन ऑफ होते हुए दिखना चाहिए। अगर ऐसा नहीं होता है तो सुनिश्चित करें कि आपने सर्किट को सही तरीके से जोड़ा है।



अगले पृष्ठ पर लिखे कोड को पूरी तरह से पढ़ें। यह मददगार होगा।



प्रोजेक्ट १

एलईडी को नियंत्रित करना

एक एलईडी को एक सेकंड के लिए चालू करें, एक सेकंड के लिए बंद करें और हमेशा के लिए दोहराते जाएँ। इस कोड को निर्माण देव द्वारा लिखा गया है, यह कोड इस्तेमाल के लिए पूरी तरह से निशुल्क है। */

//जेनुइनो एक प्रकार का कंप्यूटर है जो उन प्रोग्राम को चलाता है जिन्हें

// "स्केच" कहा जाता है। यह टेक्स्ट फाइल होते हैं जिन्हें निर्देशों का इस्तेमाल करके लिखा जाता है// जिन्हें कंप्यूटर समझ सकता है। यह एक स्केच है।

//स्केच में कोड होते हैं और साथ ही

// "कमेंट्स" भी होते हैं जो समझाते हैं कि कोड क्या करता है।

यह एक कमेंट है। "///" इसके बाद लाइन पर कुछ भी हो वो नज़रअंदाज़ कर दिया जाता है

//कंप्यूटर द्वारा।

*/ यह भी एक कमेंट है-यह कई लाइनों का हो सकता है, मगर इसे एक स्टार और स्लैश के साथ ही प्रारंभ और अंत होना चाहिए */

// एक "फंक्शन" एक कोड का ब्लॉक है जो कुछ विशेष कार्य करता है।

// सभी जेनुइनो स्केच को दो विशेष कार्य होते हैं जिन्हें

// "सेटअप" और "लूप" कहा जाता है। जेनुइनो इन कार्यों को करता है

// जब वो स्वचालित हो जाता है।

// सेटअप फंक्शन () तभी चालू होता है जब स्केच शुरू होता है।

वोइड सेटअप ()

{

कोड अगले पृष्ठ पर जारी है



// जेनुइनो में १३ डिजिटल इनपुट आउटपुट पिन होते हैं।
// आप पिन मोड फंक्शन का इस्तेमाल करके उसे बदल सकते हैं
// पिन मोड फंक्शन दो मानों को लेता है, जिसे आप टाइप करते हैं
// फंक्शन नाम के बाद उपवाक्य। पहला मान है
// एक पिन नंबर, दूसरा मान इनपुट आउटपुट शब्द है।
// इसलिए, हमारे पास, पिन मोड (१३, आउटपुट) हमारे कोड में है।

पिन मोड (१३, आउटपुट);
}

// एक बार सेटअप समाप्त हो जाने के बाद, लूप () फंक्शन बार बार हमेशा के लिए चलता रहता है

वोइड लूप ()
{

// क्योंकि हमारे पास एक एलईडी पिन १३ से जुड़ा हुआ है, अगर हम उसे बनाते हैं
// आउटपुट उच्च होगा, एलईडी को वोल्टेज मिलेगा और वो जलेगा। यदि हम बनाते हैं
// कम आउटपुट, तो एलईडी को वोल्टेज नहीं मिलेगा और वो बुझ जाएगा।
// डिजिटल राइट () एक बिल्ट-इन फंक्शन है जिसका इस्तेमाल हम
// आउटपुट पिन उच्च या कम बनाने में इस्तेमाल करते हैं। यह दो मानों को लेता है, एक पिन नंबर को,
// और शब्द उच्च या कम।

डिजिटल राइट (१३ उच्च); एलईडी को जलाएं

// डिले () एक ऐसा फंक्शन है जो कुछ समय के लिए रोकता है

// यह एक मान को लेता है, जितने समय के लिए इंतज़ार करना है, जिसे मिलीसेकंड में मापा जाता है।



कोड अगले पृष्ठ पर जारी है

```
डिले (१०००);           // एक सेकंड इंतज़ार करें  
डिजिटल राइट (१३, लो); // एलईडी बंद कर दें  
डिले (१०००);           // एक सेकंड इंतज़ार करें
```

```
// अतः ऊपर दिए गए कोड एलईडी को ऑन ऑफ करता है। हमेशा, हमेशा, हमेशा के लिए।  
}
```



आज के समय में टीवी, फ़ोन
एवं कई अन्य उपकरणों में
एलईडी इंडिकेटर होते हैं....



फेयरी लाइट्स

वो चीज़ें जिनकी आपको ज़रूरत

इस साहसिक कार्य के लिए आपको जिन चीज़ों की आवश्यकता होगी यहाँ हैं वो!



2 X जम्पर वायर



1 X एलईडी



1 X जेनुइनो बोर्ड



1 X ब्रेडबोर्ड



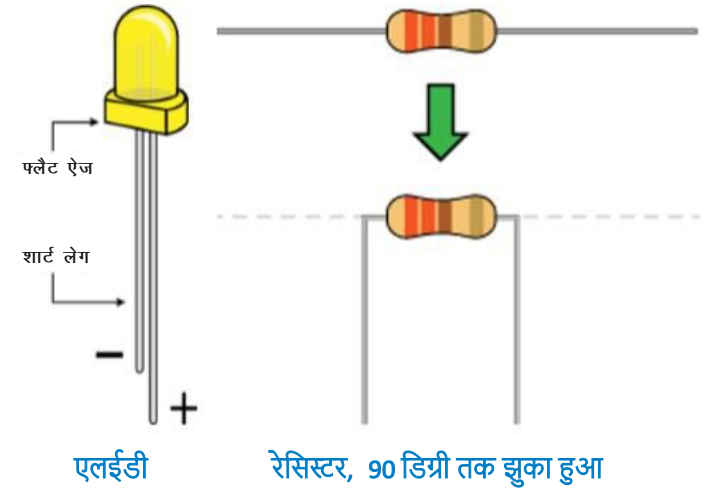
1 X रेसिस्टर 330 ओम

धागों को जोड़ना

चीजों को थोड़ा सा मज़ेदार बनाने का समय है। अब जबकि आपने एक एलईडी के साथ काम किया है, अब हम 8 एलईडी के साथ काम करने की कोशिश करेंगे। जब आप ऐसा करें तो ध्यान रखें कि सारे रेसिस्टर्स ज़मीन से जुड़े हुए हों। यह आपको अपने स्वयं के कार्यक्रमों को लिखने के लिए प्रेरित करने के लिए एक बड़ा संकेत है।
फंक्शन जैसे यह

```
for() loops and arrays[ ]
```

काम में आ सकते हैं। क्योंकि लूप एक कोड को कई बार चलाएगा, जबकि ऐरे वेरिबल्स को समूह में डालकर उन्हें प्रबंधित करना आसान बनाएगा। एक बार फिर, आपको एलईडी और रेसिस्टर को समायोजित करना होगा जिससे कि वो ब्रेड बोर्ड में घुस जाए।



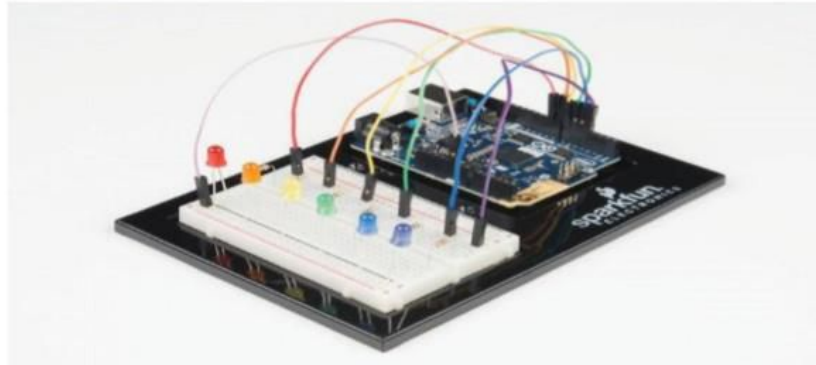


आपको आठ एलईडी एवं आठ 330 ओम की आवश्यकता होगी (नारंगी- भूरा- नारंगी)।

प्रत्येक एलईडी के लिए नेगेटिव सिरे (छोटे पाँव) को एक 330 ओम रेसिस्टर के साथ जोड़ें।

रेसिस्टर के दूसरे सिरे को GND के साथ जोड़ें।

एलईडी के पॉजिटिव सिरे (लंबे पाँव) को 9 के ज़रिये आर्डुइनो डिजिटल पिन 2 के साथ जोड़ें।



आपका बोर्ड कुछ इस तरह का दिखना चाहिए!



कोड लिखना

जेनुइनो आर्डुइनो आईडीई को खेलें और इस प्रोजेक्ट को कोड करने के लिए तैयार हो जाएँ! कई सारे तत्वों को प्रबंधित करने के लिए आपको एक ऐरे की आवश्यकता होगी। यह आपको तत्वों को समूह में लाने में मदद करता है। हम ऐरे का इस्तेमाल करके हमारे वेरिएबल्स को समूह में लाएंगे और इस समूह को यह नाम देंगे; लेड पिनस

```
int Ledpins [ ] = {2, 3, 4, 5, 6, 7, 8, 9} ;
```

हम किसी तत्व को उसकी स्थिति या उसके स्थान से संदर्भित करते हैं। शून्य से शुरू करते हुए। तो '2' का पोजीशन= 0, '3' का पोजीशन= 1 और आगे भी इसी तरह। यहाँ लेड पिनस [X] में X एक तत्व स्थिति को संदर्भित करेगा। आइए पिन '2' को हाई के रूप में सेट करें।

```
digital Write (ledpins [0], High);
```

नीरस गतिविधि को अनदेखा करने के लिए, हम यादृच्छिक गतिविधि के संचालन के लिए जेनुइनो को प्रोग्राम कर सकते हैं।

```
Index= Random (8);
```



अगले पेज पर दिए गए कोड को पूरा पढ़ें। यह मददगार होगा।

प्रोजेक्ट 2

अधिक एलईडी काम!

एलईडी को और मज़ेदार बनाएं!

यह कोड निर्माण दवे द्वारा लिखा गया है, यह कोड इस्तेमाल के लिए बिल्कुल निशुल्क है!*/

```
// एक समूह में हमारे तत्वों (पिन) को प्रबंधित करने के लिए एक ऐरे बनाना
```

```
// समूह का नाम एलईडी पिन होगा
```

```
int led pins = {2,3,4,5,6,7,8,9};
```

```
// लूप के लिए परिभाषित करना वोइड सेटअप ()
```

```
{
```

```
इंट इंडेक्स
```

```
// लूप के लिए नीचे एक कोड हमेशा के लिए चलता है, अगर यह सही है
```

```
// यहाँ लूप तब तक चलेगा जब तक और पूर्णांक मान नहीं होगा
```

```
// इंडेक्स 7 से कम या इसके बराबर होगा।
```

```
// ++ चिन्ह इंडेक्स के पूर्णांक मान में एक संख्या को जोड़ने का संकेत देता है।
```

```
for [index=0; index, <=7; index ++]
```

```
{
```

```
pinMode(ledPins[Index], OUTPUT);
```

```
}
```

```
}
```

```
// बाद में उपयोग के लिए एक और एक के बाद लूप को परिभाषित करना। यह
```

```
// एक पैटर्न में एलईडी को चालू और बंद करेगा।
```

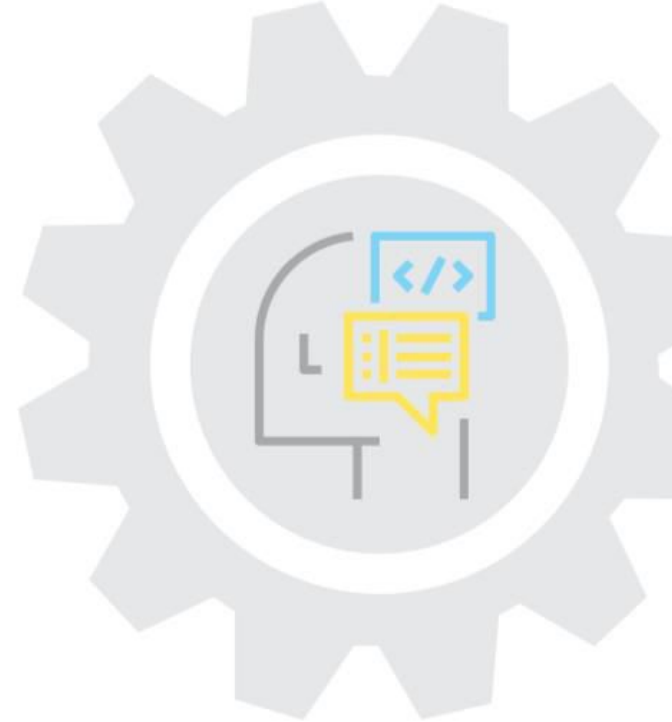
```
वोइड लूप ()
```

```
{
```

```
एक के बाद दूसर लूप () \\ एक के बाद एक के रूप में एक और कोई लूप,
```

```
\\ लेकिन बहुत कम टाइपिंग के साथ
```

```
}
```



कोड अगले पृष्ठ पर जारी है



```
// एक के बाद एक लूप काम में आता है क्योंकि  
// इसमें लूप के लिए की तुलना में कम टाइपिंग की आवश्यकता होती है।
```

```
वोइड एक के बाद एक लूप ()  
{  
  इंट इंडेक्स  
  इंट डिले टाइम =100  
  याद रखें कि इसके लिए डिले टाइम (देर करने का समय) माइक्रोसेकंड का है (इंडेक्स=0 इंडेक्स = <=7 इंडेक्स ++)  
  {  
    // हाई एलईडी के जले होने का संकेत देता है क्योंकि 5 वोल्ट का वोल्टेज डिजिटल राइट (लीडपिन [इंडेक्स], हाई) से गुजरता है;  
    डिले (डिले टाइम);  
  }  
  फॉर {इंडेक्स = 7 ; इंडेक्स >=0 , इंडेक्स --}  
  // कम एलईडी के बंद होने का संकेत देता है क्योंकि 0 वोल्ट का वोल्टेज डिजिटल राइट (लीडपिन [इंडेक्स], लो या कम) से  
  गुजरता है;  
  डिले (डिले टाइम);  
}  
// आपके एलईडी को और मज़ेदार बनाने के लिए यह वो कोड था!  
}
```



स्कॉलिंग मार्क डिस्प्ले आमतौर
पर कई एलईडी के बने होते हैं!



पुश बटन



वो चीज़ें जिनकी आपको ज़रूरत

यह वो चीज़ें हैं जिनकी आपको इस कार्य या एडवेंचर में ज़रूरत पड़ेगी!



7 X जम्पर
वायर



1 X एलईडी



1 X जेनुइनो
बोर्ड



1 X
ब्रेडबोर्ड



1x रेसिस्टर
330 ओम



2x रेसिस्टर
10k ओम



2x
पुश बटन

धागों को जोड़ना

अब तक हमने उन परियोजनाओं की बात की है जो आउटपुट से संबंधित है। अब यहाँ एक प्रोजेक्ट है जो इनपुट से संबंधित है और मेरा यकीन कीजिए यह बेहद कूल है!

जिस तरह से जेनुइनों के साथ एक पुश बटन काम करता है वह यह है कि जब बटन को दबाया जाता है, तो वोल्टेज कम होता है। जेनुइनों इसे पढ़ता है और तदनुसार प्रतिक्रिया करता है।

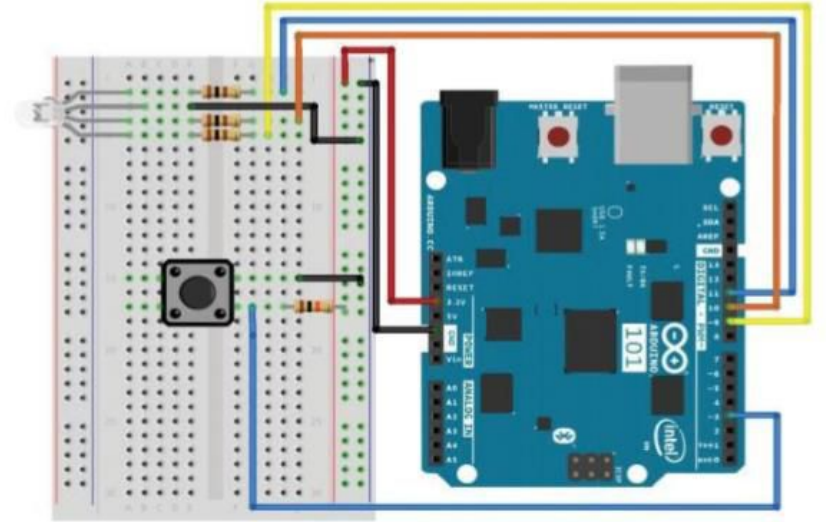
जब बटन दबाया नहीं जाता है तो वोल्टेज को ऊंचा रखने के लिए पुल अप रेसिस्टर या प्रतिरोधक का भी उपयोग किया जाएगा।

पुश बटन्स

पुश बटन को हुक करने का सबसे आसान तरीका

दो तारों को किसी भी विपरीत कोनों से जोड़ना है। ग्राउंड [GND] पर पुश बटन १ से किसी भी पिन को कनेक्ट करें।

पुशबटन के विपरीत विकर्ण पिन को डिजिटल पिन २ से कनेक्ट करें।





पुशबटन 2 से किसी भी पिन को ज़मीन [GND] से जोड़ें। पुशबटन के विपरीत विकर्ण पिन को डिजिटल पिन 3 से कनेक्ट करें।

डिजिटल पिन 2 और 3 एवं GND के बीच 10k प्रतिरोधकों (भूरा / काला / लाल) को भी जोड़ें। इन्हें "पुल्ल उप रेसिस्टर्स" भी कहा जाता है। वे यह सुनिश्चित करते हैं कि इनपुट पिन या तो 5 वोल्ट (न दबाया गया) या GND (दबाया गया) होगा, और बीच में कहीं नहीं होगा।

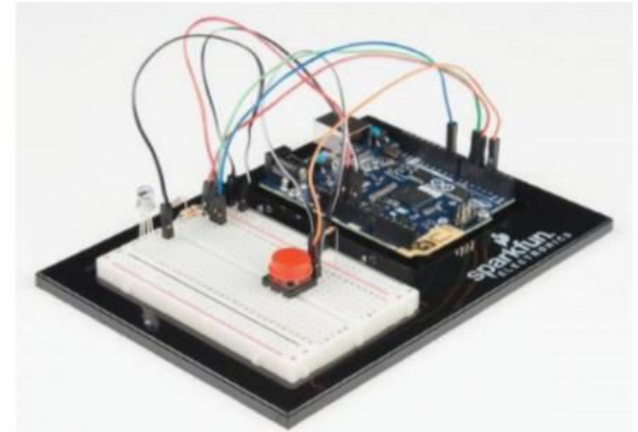
(याद रखें कि एनालॉग इनपुट के विपरीत, डिजिटल इनपुट केवल हाई या लो हैं।)

एलईडी

यूनो सहित अधिकांश आर्डुइनो में पहले से ही एक एलईडी और प्रतिरोधक या रेसिस्टर है जो पिन 13 से जुड़ा है, आपको किसी भी अतिरिक्त सर्किटरी की आवश्यकता नहीं है।

लेकिन अगर आप पिन 13 को किसी दूसरी एलईडी के साथ जोड़ना चाहते हैं, तो अपने एलईडी के पॉजिटिव सिरे को आर्डुइनो डिजिटल पिन 13 से जोड़ें। अपने एलईडी के नेगेटिव सिरे को 330 ओम अवरोधक से जोड़ें।

प्रतिरोधक के दूसरे हिस्से को जमीन से जोड़ें



आपका बोर्ड कुछ इस प्रकार दिखेगा!

कोड लिखना

जेनुइनो आर्डुइनो आईडीई को खोलें और इस प्रोजेक्ट को कोड करने के लिए तैयार हो जाएँ! हमें जेनुइनो को यह समझाने की आवश्यकता होगी कि जिस डिजिटल इनपुट का हम उपयोग कर रहे हैं वह इनपुट कि तरह इस्तेमाल हो रहा है या आउटपुट की तरह। नीचे दिया गया कोड वह करता है।

पिन मोड (बटन 2 पिन, इनपुट);

डिजिटल इनपुट को पढ़ने के लिए, हम डिजिटल के साथ इस कोड का उपयोग करते हैं।

बटन 1 स्टेट = डिजिटल रीड (बटन 1 पिन);

क्योंकि हमने बटन को GND से जोड़ दिया है, उसे दबाए जाने पर कम दिखाना पड़ेगा। नीचे दिया गया कोड यह काम करेगा।

यदि (बटन 1 स्टेट == लो)



अगले पृष्ठ पर दिए गए
कोड को पूरा पढ़ें।
यह मदद करेगा।



/* प्रोजेक्ट 3

पुश बटन!

बचाने के लिए बटन्स!

इस कोड को निर्माण दवे ने लिखा है। यह कोड उपयोग के लिए निशुल्क उपलब्ध है! */

//सबसे पहले हम पिन नंबरों के लिए कांस्टेंट तय करेंगे।

// इससे कोड का पालन करना आसान हो जाएगा।

Look at the code on English (Constint button..... = 13; एलईडी पिन

वोइड सेटअप ()

{

पुश बटन पिन को इनपुट के तौर पर कमांड दें (Look at the formula in English: pin mode button....Input);

// एलईडी पिन को एक आउटपुट होने के लिए कमांड दें (Look at the formula in English: pin mode button....Output);

}

वोइड लूप ()

{

(Look at the formula in English: int button.....State;) //

पुश बटन की अवस्था को रोकने के लिए वेरिबल्स

// क्योंकि एक पुश बटन में केवल दो स्टेट या अवस्थाएं होती हैं, हम इसे डिजिटल रीड के तौर पर इस्तेमाल कर सकते हैं

// फंक्शन। और हम वर्तमान पुश बटन अवस्था को

// दो वेरिबल्स के रूप में पढ़ेंगे।

(Refer to the formula in English)

// यदि बटन को दबाया जाता है तो यह

// GND से जुड़ जाएगा। यदि बटन को दबाया नहीं जाता है,

// तो पुल अप रेसिस्टर इसे 5 वोल्ट के साथ जोड़ देगा।

// तो अवस्था कम की रहेगी, जब इसे दबाया जाएगा,



कोड अगले पृष्ठ पर जारी है



```
if((button1State == HIGH) && (button2State == LOW))  
  
    // यदि हम बटन 1 या 2 को दबायेंगे  
    // if we're pushing button 1 OR button 2  
  
    && ! // AND we're NOT  
    (Look at the formula) // और यदि हम नहीं  
    ((button1State == LOW) && (button2State == LOW))) // pushing button 1 AND button 2  
    (Look at the formula)  
    // then...  
    // तो.....  
    {  
    (Look at the formula) // एलईडी को जलेगा  
    digitalWrite(ledPin, HIGH); // turn the LED on  
    }अन्यथा  
    else  
    { (Look at the formula in English) //एलईडी बुझ जाएगा  
    // वैसे पुश बटन के लिए यह कोड था  
    digitalWrite(ledPin, LOW); // turn the LED off  
    }  
    }  
    // Well, that was the code for Push button.  
}  
}
```



खैर, मैं आपको एक पुश बटन के उपयोग के बारे में नहीं बता रहा हूँ!



धन्यवाद

