

दीप – देवनागरी सक्षम प्रोग्रामिंग

आता प्रोग्रामिंग करा मराठीत !

(माहिती पुस्तिका)

प्रकल्प सदस्य:

शुभम अशोक हाळ्ळे

मयूर संदीप जगताप

रुचिता संजीव कोलते

निकिता सुधीर बाभळे

मार्गदर्शन:

प्रा. अदिती एस जहागीरदार

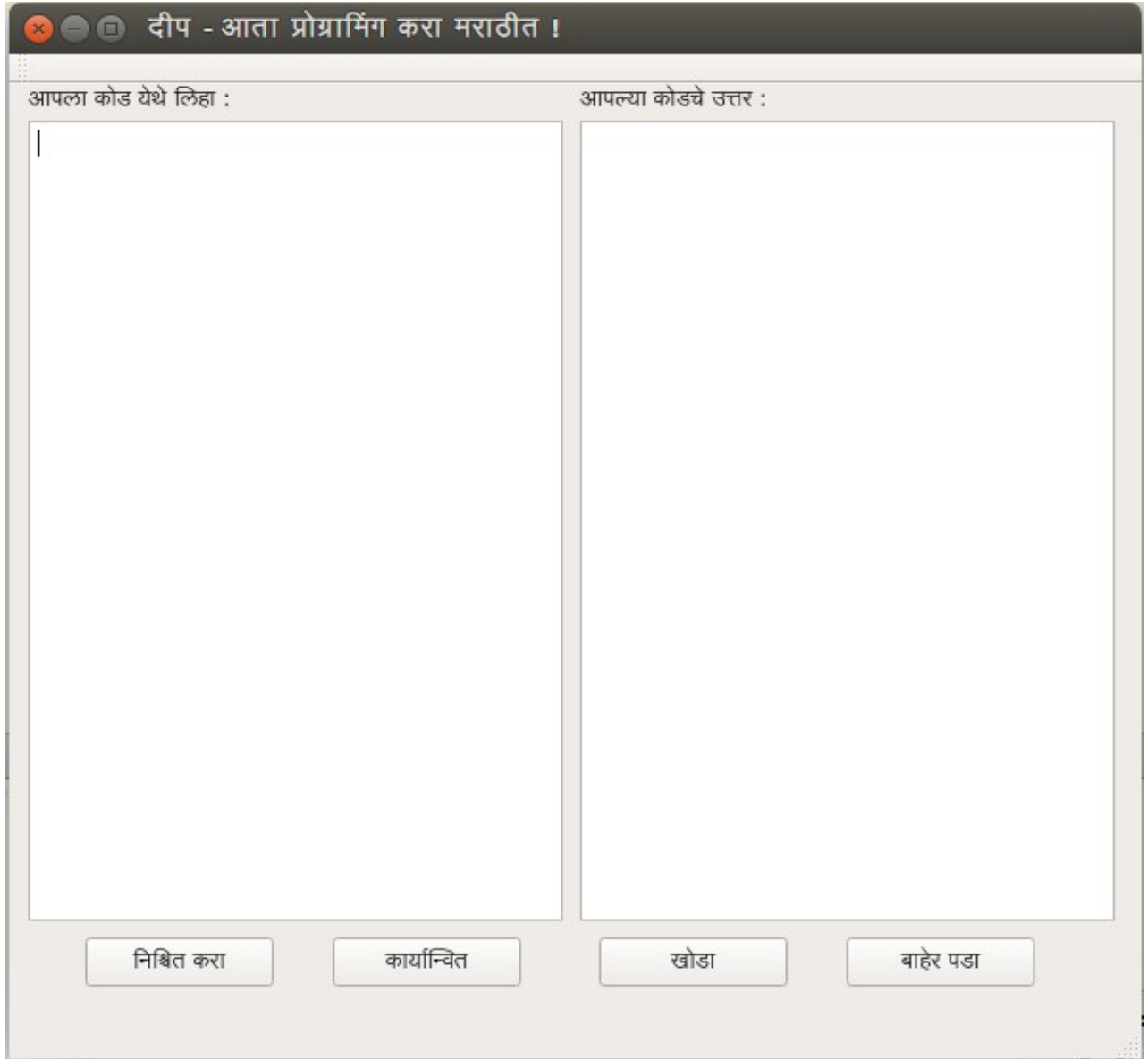
अणुक्रमणिका

१. दीप – परिचय
२. दीप कामकाज विंडो
३. प्रवाह-नियंत्रण विधान
 - ३.१ जर विधान, नाहीतर, किंवाजर
 - ३.२ साठी विधान
 - ३.३ जोपर्यंत विधा
४. कृती
 - ४.१ माहिती कृती
 - ४.२ छापा कृती
५. चुका आणि अपवाद
 - ५.१ वाक्यरचना चुका
 - ५.२ इतर अनपेक्षित चुका

१. परिचय दीप

विविध माध्यमांच्या संशोधनातून समोर आले आहे कि, मातृभाषा ही मानसिक विकासासाठी शिक्षणात मध्यवर्ती भूमिका बजावत असते. दीप हे सॉफ्टवेअर अश्या लोकांसाठी आहे ज्यांना मराठी चांगल्या पद्धतीने येते, पण इंग्रजीचा वापर हा जाचक वाटतो. दीप एक नवीन प्रोग्रामिंग भाषा नाहीये पण दीप हे एक सॉफ्टवेअर आहे जे वापरकर्त्यांना मराठीतून पायथन प्रोग्राम लिहिण्यास प्रोत्साहन देते. हे सॉफ्टवेअर Python 2.7 वर कार्य करते. यांच्यात एक वापरकर्त्यांना अनुकूल असे GUI म्हणजेच विंडो आहे आणि यात सर्व सूचना मराठीत आहेत. त्यामुळे, मराठीतून सहज कोडींग अंमलबजावणी करण्यासाठी आणि ते जाणून घेण्यासाठी वापरकर्त्यांना सक्षम करते.

दीप – वापरण्यासाठी विंडो :



१. जतन करा बटण

प्रथम वापरकर्त्यांनी आपला मराठी प्रोग्राम येथे लिहिणे अपेक्षित आहे. त्यानंतर ते जतन करण्यासाठी "जतन करा" या बटणावर क्लिक करणे अपेक्षित आहे.

! टिप : जर वापरकर्त्याने जतन न करता प्रोग्राम कार्यान्वित केला तर अनपेक्षित प्रोग्राम कार्यान्वित होण्याची शक्यता आहे.

२. कार्यान्वित बटण

प्रोग्राम जतन केल्यानंतर हे बटण क्लिक करा. तो प्रोग्राम कार्यान्वित करतो आणि परिणाम दाखवतो. जर काही त्रुटी असेल तर हे बटण दाबल्यानंतर त्रुटी प्रदर्शित होतील.

३. खोडा बटण

हे बटण दाबून इनपुट आणि आऊटपुट दोन्ही क्षेत्र साफ होईल.

४. बाहेर पडा बटण

हे बटण दाबून GUI बंद होते.

५. येथे कोड लिहा

मराठी कोड येथे लिहिणे अपेक्षित आहे..

६. आउटपुट येथे प्रदर्शित केले जाते

आउटपुट कोड येथे प्रदर्शित केला आहे. जर त्रुटी असतील, तर ते देखील या क्षेत्रात प्रदर्शित केल्या जातील.

३. प्रवाह-नियंत्रण विधाने

विधान ३.१: साठी विधान

काही वाक्यरचना पुन्हा पुन्हा कार्यान्वित करावयाच्या असतात, त्याकरिता साठी हे विधान वापरले जाते.

साठी विधानानंतर लागोलाग वाक्यांची रचना असणे आवश्यक आहे.

वाक्यरचना (मांडणी) :

साठी <अक्षर नाव> आत मध्ये (<श्रेणी>):

उदाहरण:

साठी अ आत मध्ये (०, १०):

विधान ३.२: जर विधान

काही वाक्यरचनांना कार्यान्वित करण्यासाठी ठराविक अटींची पूर्तता करणे गरजेचे आहे. त्या करिता "जर" विधान वापरले जाऊ शकते.

वाक्यरचना :

जर (अट):

<वाक्ये>

किंवाजर (अट):

<वाक्ये>

नाहीतर:

<वाक्ये>

उदाहरण:

जर (अ<१००):

छापा अ

किंवाजर (अ>१०००):

छापा "आहे"

नाहीतर:

छापा "नाही"

३.३: जोपर्यंत विधान

जोपर्यंत एखादी अट सत्य आहे, तोपर्यंत वाक्ये कार्यन्वित होतात.

वाक्यरचना (मांडणी) :

जोपर्यंत(<अट>):

उदाहरण:

जोपर्यंत(अ<१०):

छापा अ

अ = अ+१

૪. કૃતી વિધાને

૪.૧: માહિતી કૃતી વિધાન

વાપરકર્ત્યાકિઢૂન ઇનપુટ ઘેણ્યાસાઠી માહિતી હી કૃતી વાપરતા યેઈલ.

વાક્યરચના (માંડણી) :

માહિતી ("<વિચારા>")

ઉદાહરણ:

માહિતી ("કૃપયા અ ચી કિંમત ટાકા")

!ટિપ – સંખ્યા ઇનપુટ ઘેણ્યાસાઠી સંખ્યા(માહિતી("<વિચારા>")) હી રચના વાપરાવી.

વાક્યરચના (માંડણી) :

માહિતી ("<વિચારા>")

ઉદાહરણ:

માહિતી ("કૃપયા અ ચી કિંમત ટાકા")

४.२: छापा कृती विधान

आउटपुट छापण्याकरिता हे विधान वापरले जाऊ शकते.

वाक्यरचना (मांडणी) :

छापा <variable name>किंवा" <any message>"

उदाहरण:

छापा "नमस्कार"

५. चुका आणि अपवाद

५.१: चुकीची मांडणी

वरील पैकी कोणतीही वाक्यरचना जर जुळली नाही तर "चुकीची मांडणी" आणि त्यासोबत ओळ क्रमांक देखील प्रदर्शित केला जाईल.

चुक : चुकीची मांडणी – ओळ क्रमांक १०